Universidade de Brasília
Instituto de Ciências Exatas
Departamento de Ciência da Computação

Detecção de objetos para identificar espécimes de macaúba e babaçu por meio de imagens de drone utilizando as arquiteturas YOLOv4 e YOLOv9

Pedro Lucas Silva Haga Torres

Monografia apresentada como requisito parcial
para conclusão do Curso de Computação — Licenciatura

Orientador
Prof. Dr. Díbio Leandro Borges

Brasília
2024

Universidade de Brasília
Instituto de Ciências Exatas
Departamento de Ciência da Computação

Detecção de objetos para identificar espécimes de macaúba e babaçu por meio de imagens de drone utilizando as arquiteturas YOLOv4 e YOLOv9

Pedro Lucas Silva Haga Torres

Monografia apresentada como requisito parcial
para conclusão do Curso de Computação — Licenciatura

Prof. Dr. Díbio Leandro Borges (Orientador)
CIC/UnB

Prof. Dr. Camilo Chang Dórea
Universidade de Brasília

Prof. Dr. Edson Eyji Sano
Embrapa Cerrados

Prof. Dr. Jorge Henrique Cabral Fernandes
Coordenador do Curso de Computação — Licenciatura

Brasília, 16 de setembro de 2024

Dedicatória

Dedico este trabalho aos meus pais, família, namorada e amigos. Em especial, aos meus pais, por todo o esforço e dedicação que empenharam para me criar. Decerto, eles foram os gigantes sobre cujos ombros pude me apoiar para ver mais longe.

Agradecimentos

Agradeço ao pessoal da Embrapa pela disponibilização das imagens de drone que foram essenciais a essa pesquisa. Em particular, à servidora Gabriela Silva Ribeiro que foi um importante ponto de contato e ao doutor Edson Eyji Sano que compartilhou sua pesquisa além das imagens em si.

Também gostaria de agradecer à Universidade de Brasília, sobretudo aos bons professores e professoras que tive em minha carreira acadêmica e aos servidores técnico-administrativos, que desempenham um papel fundamental no funcionamento de nossa universidade.

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES), por meio do Acesso ao Portal de Periódicos.

Resumo

A macaúba (*Acrocomia intumenscens*) tem potencial para se tornar um recurso de grande valor no *framework* da economia sustentável no Brasil. Neste trabalho foram desenvolvidas duas soluções para detecção de palmeiras macaúba e babaçu (*Attalea speciosa*) em ambientes naturais por meio de imagens obtidas a partir de UAV e algoritmos de *deep learning*. As soluções se baseiam nas arquiteturas YOLOv4 e YOLOv9, as quais obtiveram mAP@0.50 de $75,63\% \pm 3,29\%$ (média $\pm$ desvio padrão) e $72,7\% \pm 2,3\%$, respectivamente, avaliadas pelo processo de validação cruzada *k-fold*. Os resultados obtidos apontam para a facilidade em distinguir essas espécies por meio da metodologia proposta, contudo, podem ser feitas melhorias no processo de anotação das imagens, ou na divisão dos ortomosaicos para produção dessas imagens utilizadas no treinamento dos algoritmos utilizados.

Palavras-chave: Sensoriamento remoto, detecção de objetos, aprendizagem profunda, YOLO, UAV

Abstract

The macauba palm (*Acrocomia intumenscens*) has potential to become a high-value resource within the framework of Brazil's sustainable economy. This study evaluates two convolutional neural networks (CNN) architectures for the tasks of identifying and differentiating specimens of macauba and babassu (*Attalea speciosa*) in their natural environments in the context of object detection and UAV-based imagery. The evaluated CNN architectures were YOLOv4 and YOLOv9, which achieved an mAP score of $75.63\% \pm 3.29\%$ (average $\pm$ standard deviation) and $72.7\% \pm 2.3\%$, respectively, as evaluated through k-fold cross-validation. The results indicate the effectiveness of the proposed methodology in distinguishing these species, although there are improvements to be made in the image annotation process, or the process of dividing the orthomosaics that produced the images used in the training phase.

Keywords: Remote sensing, object detection, deep learning, YOLO, UAV

Sumário

Lista de Figuras

Lista de Tabelas

Lista de Abreviaturas e Siglas

API do inglês, *aplication programing interface*, interface de programação de aplicação.

BB do inglês, *bounding box*, caixa delimitadora.

CNN do inglês, *convolutional neural networks*, redes neurais convolucionais.

curva PR curva precisão-revocação.

DL do inglês, *deep learning*, aprendizagem profunda.

FPS do inglês, *frames per second*, quadros por segundo.

INMET Instituto Nacional de Meteorologia.

LISA Laboratório de Imagens, Sinais e Acústica vinculado ao Departamento de Ciência da Computação da Universidade de Brasília.

ML do inglês, *machine learning*, aprendizagem de máquina.

MS COCO do inglês, Microsoft *Common Objects in Context*, é um conjunto de dados produzido para o contexto de problemas de detecção de objetos.

SGD do inglês, *stochastic gradient descent*, gradiente descendente estocástico.

UAV do inglês, *unmanned aerial vehicles*, veículos aéreos não-tripulados.

Capítulo 1

Introdução

A biodiversidade brasileira é amplamente reconhecida e esforços significativos têm sido empreendidos para explorar seus recursos de maneira sustentável, visando o desenvolvimento econômico das populações locais sem comprometer o meio ambiente. Dentre as espécies vegetais com alto potencial econômico, destacam-se as palmeiras oleaginosas, especialmente a macaúba (*Acrocomia intumescens*) e o babaçu (*Attalea speciosa*), encontradas em grande quantidade na região Nordeste do Brasil [1, 2]. Apesar de serem exploradas localmente com pouca tecnologia, a macaúba tem ganhado importância como fonte de óleos vegetais, sendo vista como uma peça-chave para a política de bioeconomia [3]. Ainda que iniciativas para o cultivo em larga escala da macaúba estejam em andamento no Brasil [4, 5], é provável que leve algum tempo e necessite de mais investimentos até que uma produção significativa seja alcançada, capaz de fornecer matérias-primas de maneira confiável. Nesse período, o aproveitamento das populações naturais de macaúba pode ser uma alternativa para agregar valor econômico, especialmente para as comunidades locais.

Atualmente, os frutos da macaúba são colhidos em pequena escala e de forma desorganizada, apesar da abundância da espécie [6, 2]. No entanto, há grande potencial para que essas populações nativas se tornem uma fonte sustentável de frutos e serviços ambientais. A macaúba é uma planta versátil, oferecendo desde alimentos e óleos vegetais diversificados até biomassa para biocombustíveis sólidos e líquidos, além de inúmeros produtos de base biológica [7, 8]. Sua produção de óleo é alta, comparável à da palmeira africana de dendê (*Elaeis guineensis*), a principal *commodity* do setor. Entretanto, a macaúba possui a vantagem de crescer em áreas não adequadas para o dendê, como em biomas fora das florestas tropicais. A produtividade de óleo da macaúba cultivada no Brasil, com densidade de 400 árvores por hectare, é estimada em cerca de 4.000 toneladas por ano, superando as produtividades do babaçu, soja e algodão [1].

A distribuição da macaúba abrange toda a América tropical e subtropical, desde o sul do México até o Paraguai e Argentina. No Brasil, a espécie ocorre desde a região

Norte do país até o estado do Paraná, principalmente em áreas abertas e relativamente secas [9, 10]. A presença de ecótipos de macaúba em regiões com escassez hídrica sugere seu potencial para cultivo em larga escala em áreas marginais, como no semiárido do Nordeste, que possui baixos Índices de Desenvolvimento Humano. Além disso, a macaúba apresenta potencial para recuperação de áreas degradadas, como os 70 milhões de hectares de pastagens degradadas no Brasil, contribuindo para a descarbonização da economia. O cultivo intercalado da macaúba com outras culturas anuais ou perenes, ou mesmo com a pecuária, pode intensificar e diversificar a produção agrícola, como demonstrado em estudos que sugerem o consórcio de café e macaúba como uma estratégia para mitigar as mudanças climáticas [11].

1.1 Sensoriamento remoto para identificação de palmeiras

O uso de técnicas de sensoriamento remoto em conjunto com métodos de aprendizado de máquina pode facilitar a identificação e exploração sustentável das áreas com grandes concentrações de macaúba. Estas técnicas contrastam com os métodos tradicionais de observação e contagem de árvores *in loco*, oferecendo maior eficiência e alcance com custos reduzidos. Imagens de satélite, frequentemente usadas devido à sua disponibilidade e baixo custo, apresentam desafios como baixa resolução espacial e sensibilidade às condições climáticas. Contudo, têm-se obtido bons resultados na detecção de palmeiras utilizando essa tecnologia [12, 13, 14, 15, 16].

O sensoriamento remoto por meio de veículos aéreos não-tripulados (UAV, do inglês, *unmanned aerial vehicles*), como drones, tem ganhado destaque em aplicações e na pesquisa. UAV possuem custo relativamente baixo de aquisição, oferecem alta resolução espacial, são capazes de capturar dados precisos de geolocalização, conseguem alcançar áreas de difícil acesso e são menos afetados por condições meteorológicas adversas. No entanto, essa abordagem também possui limitações, como a menor cobertura de área em comparação a satélites e aviões tripulados e suscetibilidade a ventos fortes. Existem exemplos bem-sucedidos do uso de imagens obtidas a partir de drones para identificar palmeiras [13, 17, 18, 19, 20].

1.2 Detecção de objetos em imagens utilizando aprendizagem profunda e redes neurais convolucionais

Algoritmos de aprendizagem de máquina (ML, do inglês, *machine learning*) permitem uma análise automatizada de grandes volumes de dados. Dentro dessa área, redes neurais convolucionais (CNN, do inglês, *convolutional neural networks*) são amplamente utilizadas para detecção de objetos em imagens e vídeos [21]. O termo "detecção de objetos" refere-se a tarefa de localizar e classificar múltiplos objetos em uma imagem. Existem diferentes variantes do problema, como segmentação semântica, na qual cada pixel da imagem deve ser classificado como pertencente a uma classe, ou a segmentação de instâncias, onde objetos diferentes, mas pertencentes a uma mesma classe, recebem seu próprio rótulo. Detecção de objetos, neste trabalho, refere-se a tarefa de localizar e identificar as espécies (ou classes, como referida no jargão de ML) de macaúba e babaçu por meio de caixas delimitadoras (BB, do inglês, *bounding boxes*). A capacidade de diferenciar entre ambas espécies também é importante no contexto deste trabalho, uma vez que elas são semelhantes em termos de espectro, tamanho e formato.

O aprendizado profundo (DL, do inglês, *deep learning*) é uma área de ML que se sobressai em problemas simples e intuitivos para humanos, porém, difíceis de serem descritos por meio de formalizações matemáticas, como o reconhecimento de fala ou de rostos de pessoas [22]. Como comentado anteriormente, CNNs são amplamente utilizadas para a tarefa de detecção de objetos, e possuem em sua base filtros de convolução que são utilizados para encontrar e extrair características relevantes dos dados. O treinamento de um modelo de detecção de objetos baseado em CNN consiste em encontrar os filtros adequados para extrair as características relevantes de uma imagem de modo a localizar e classificar potenciais objetos. Entretanto, esse tipo de abordagem exige grandes quantidades de dados para conseguir gerar uma solução adequada e generalizável [23].

1.3 Estrutura do trabalho

O presente trabalho apresenta a revisão de literatura, metodologia e resultados experimentais para o desenvolvimento e treinamento de uma solução de ML orientada a detectar e diferenciar espécimes de macaúba e babaçu na região Nordeste do Brasil. A solução proposta faz uso de imagens capturadas a partir de UAV, editadas na forma de um ortomosaico e alimentadas em duas arquiteturas de DL orientadas a detecção de objetos: a YOLOv4 [24] e a YOLOv9 [25]. Por fim, foi feita a análise de viabilidade e potenciais limitações para a solução proposta. A estrutura deste trabalho segue: o Capítulo 2 contém a revisão de literatura, seguido pelo Capítulo 3 no qual descreve-se a metodologia a ser

adotada, o Capítulo 4 refere-se ao resultados experimentais obtidos e o Capítulo 5 expõe a conclusão desta monografia.

Capítulo 2

Revisão da literatura

Este capítulo trata da revisão da literatura conduzida. Inicia-se expondo a dificuldade de se encontrar dados relativos a imagens obtidas a partir de UAV com anotações relacionadas a flora. Em seguida, comenta-se sobre trabalhos de classificação e detecção de objetos a partir de imagens obtidas de satélites e drones. Por fim, aprofunda-se nos tópicos de ML, DL, as arquiteturas da família YOLO e técnicas associadas a *machine learning*.

2.1 Conjuntos de dados

Como comentado anteriormente, soluções de DL exigem grandes quantidades de dados para alcançar resultados adequados e generalizáveis. Existem conjuntos de dados com imagens obtidas a partir de técnicas de sensoriamento remoto como em [26, 27, 28], mas poucos oferecem imagens de vegetação e, quando oferecem, não estão anotadas (*i.e.*, os espécimes de plantas não estão marcados com caixas delimitadoras nem identificadas). Xia et al. [26] compilaram um conjunto de dados a partir de imagens do Google Earth[1] extraindo 10.000 imagens aéreas com resolução de 600 × 600 px e divididas em 30 classes (como aeroporto, praia, cultivo agrícola, floresta, prado, entre outros), sendo um *dataset* mais relacionado a tarefa de classificação de imagens. O conjunto de dados DOTA de Xia et al. [27] conta com 2.806 imagens de diferentes resoluções (entre 800 × 800 a 4.000 × 4.000 px) extraídas do Google Earth e divididas em 15 categorias diferentes (como veículo grande e pequeno, helicóptero, avião, campo de futebol, quadra de tênis, entre outros), esse *dataset* possui anotações na forma de caixas delimitadoras, entretanto, não se relaciona com objeto de estudo deste trabalho. O conjunto de dados VisDrone [28] possui 10.209 imagens obtidas a partir de UAV, divididas em 6.471 para treino, 548 para validação 3.190 para teste (que é subdividido em imagens para avaliação da competição e imagens abertas para avaliação pública) e contém 10 classes (pedestre, pessoa, ônibus,

[1] `https://earth.google.com/`

carro, etc.), seus rótulos são notavelmente voltados para o ambiente urbano. Como é possível observar, conjuntos de dados abertos relacionados de maneira mais específica a natureza deste trabalho (*i.e.*, contendo marcações de espécimes da flora) são escassos na literatura.

Conjuntos de áreas correlacionadas, como a agricultura, também são de difícil acesso, pois, ou não estão publicados [29, 30], ou são proprietários [31]. Desse modo, para pesquisas na área, é necessário não apenas o esforço de obtenção das imagens, assim como o de anotação. Lin et al. [29] produziram um conjunto de dados a partir de imagens RGB capturadas sistematicamente com um UAV em diferentes altitudes (20, 30, 40 e 50 m), com resolução de 224 × 224 px e divididas em 12 classes de diferentes espécies de árvores (*Blechnum orientale*, *Carmona microphylla*, *Eucalyptus citriodora*, entre outras), totalizando 2.293 imagens sem *data augmentation* e 21.863 com. Esse conjunto de dados é supostamente público e disponibilizado na plataforma GitHub[2], entretanto, há apenas duas imagens no repositório. Morales et al. [30] coletaram imagens RGB de palmeiras de buriti (*Mauritia flexuosa*) a partir de UAV em uma região da floresta amazônica no Peru entre os anos de 2015 a 2018, totalizando 25.248 imagens de 512 × 512 px com máscaras para segmentação semântica. O conjunto de dados é disponibilizado no referido trabalho, porém, o link para o *dataset* está inacessível[3].

2.2 Detecção de palmeiras por meio de imagens de satélite

Há ampla literatura na área de detecção de palmeiras por meio de imagens de satélite, como em [12, 14, 15, 16]. Al-Helaly e Al-Helaly [12] publicaram um algoritmo que, por meio de técnicas de visão computacional, extrai o formato de palmeiras oleaginosas de modo a isolar cada espécime individualmente na imagem. Zheng et al. [14] conduziram um estudo para identificar palmeiras dentro de um ambiente de cultivo com CNN (modelo ZF [32], VGG_CNN_M_1024 [33] e VGG16 [34]) no sul da Malásia, com F1-score de 90 a 96%. Nesse último estudo, os autores utilizaram imagens obtidas em 2006 a partir do satélite QuickBird e selecionaram 14 áreas de 600 × 600 px, separando 6 para treinamento, 2 para validação e outras 6 para teste. As áreas de treinamento e validação foram convertidas para a resolução de 2.400 × 2.400 px e, no caso das áreas selecionadas para treinamento, foram feitos cortes aleatórios de dimensão 500 × 500 px, gerando um total de 5.000 amostras para o aprendizado da rede; no caso das áreas de validação, as imagens foram recortadas na dimensão de 66 × 66 px com parâmetro de sobreposição (*overlap*) não

[2] `https://github.com/FAFU-IMLab/FDNs/tree/master/Dataset`
[3] `http://didt.inictel-uni.edu.pe/dataset/MauFlex_Dataset.rar`

reportada pelos autores. Para melhorar os resultados, os autores implementaram um filtro de pós-processamento baseado em regras de plantio, elevando os resultados de F1-score de 83,58 a 96,62% para 90,09 a 97,25%, com as marcações sendo consideradas corretas para palmeiras cujas caixas delimitadoras possuíssem interseção sobre união (IoU, do inglês, *intersection over union*) maior ou igual que 0,5.

Mubin et al. [15] utilizaram uma arquitetura baseada na LeNet [35] para diferenciar palmeiras jovens e adultas em diferentes ambientes de cultivo. Para tanto, os autores utilizaram imagens RGB obtidas a partir do satélite WorldView-3, localizadas no campus da Universidade de Putra Malásia. A imagem de satélite obtida foi dividida em 24 partes com resolução de 991 $\times$ 1.155 px e subdivididas em imagens de 26 $\times$ 26 px ou 31 $\times$ 31 px para áreas contendo palmeiras jovens e adultas, respectivamente, centralizando as imagens que contém palmeiras na coroa da planta. Dessa maneira, esse trabalho consiste na classificação de imagens como sendo de palmeiras (jovens ou adultas) ou não. Nesse trabalho, foram utilizadas 2.323 imagens para a classificação de palmeiras jovens e 1.384 de palmeiras adultas, divididas nas proporções de 20-70-10% (treino, validação e teste, respectivamente) para jovens e 10-85-5% para adultas. Os autores reportam como resultado a média aritmética entre precisão e revocação, alcançando 95,11% e 92,96% para palmeiras jovens e adultas, respectivamente.

Culman et al. [16] realizaram um estudo com o objetivo de classificar e localizar palmeiras do gênero *Phoenix* por meio de segmentação semântica e imagens multiespectrais da região das Ilhas Canárias, nas quais a coroa da palmeira é menor do que um pixel de largura. As imagens foram obtidas a partir da constelação de satélites Copernicus Sentinel-2, possuem resolução espacial de 10, 20 e 60 m e 13 bandas espectrais, e os rótulos foram elaborados a partir de um estudo oficial das autoridades das Ilhas Canárias e de um esforço dos autores para identificar aproximadamente 12.500 palmeiras por meio de ortofotos e da interpretação dos autores. Os autores utilizaram uma rede baseada na arquitetura ResNet [36]. Os resultados reportados foram acurácia de 92,1%, revocação igual a 43,8% e precisão de 52,2%, atribuindo alto valor da acurácia ao fato que essa métrica leva em consideração a quantidade de negativos verdadeiros e a maioria dos pixels (96,1%) não contém palmeiras em suas imagens. Por outro lado, o estudo conseguiu demonstrar a vantagem do uso de imagens multiespectrais, uma vez que os mesmos métodos utilizados em imagens RGB não conseguiram realizar esse tipo de detecção. Os resultados de Culman et al. apontam para a dificuldade de se localizar ocorrências de palmeiras em um ambiente natural, em contraposição aos trabalhos de Mubin et al. e Zheng et al. que foram realizados em diferentes ambientes de cultivo.

2.3 Detecção de palmeiras por meio de imagens de UAV

Zortea et al. [13] desenvolveram duas CNN para identificar palmeiras (referidas apenas como "*oil-palm tree*", sem identificação da espécie) em uma plantação comercial no município de Tailândia, no estado do Pará, Brasil, com acurácia média de 88-98%. A captura das imagens foi feita utilizando um UAV modelo Echar 20c no espectro de luz visível (RGB) em três locais diferentes com área aproximada de 35 hectares, cada, e palmeiras com idade variando entre 2, 4 e 16 anos. Cada uma das CNN desenvolvidas pelos autores focam em escalas diferentes (10 e 20 cm/px) e predizem a probabilidade de que uma palmeira esteja no centro de um recorte da imagem, por fim, os autores propuseram um terceiro modelo que faz a detecção por meio da média simples entre as probabilidades calculadas pelas duas redes. Esse trabalho focou na detecção de palmeiras para contagem, ao contrário de uma detecção para marcação por meio de caixas delimitadoras. Dessa maneira, as métricas reportadas foram acurácia, precisão e revocação calculadas por meio do número de palmeiras detectadas corretamente, número total de detecções e número de palmeiras no conjunto de verdade de referência (*ground truth*). Os melhores resultados (considerando todo o conjunto de dados) foram obtidos pelo terceiro modelo, com precisão de 94,2%, revocação de 97,0% e acurácia média de 95,6%.

Ferreira et al. [17] conduziram um estudo para identificar, por meio de segmentação semântica em imagens de drone, palmeiras de açaí (*Euterpe precatoria*), jací (*Attalea butyracea*), paxiubão (*Iriartea deltoidea*) e uma quarta palmeira não identificada em uma floresta experimental da Embrapa no município de Rio Branco, no estado do Acre, Brasil. Em sua abordagem, os autores utilizaram um UAV DJI Phantom 4 Professional e a câmera do próprio drone para obter imagens RGB a uma altura de 120 m da copa da floresta, coletando um total de 1423 fotos. Os autores desenvolveram uma arquitetura baseada na DeepLabv3+ combinada com a ResNet-18, de modo a gerar *score maps* (do inglês, mapas de pontuação, em tradução livre) nos quais cada pixel contém a probabilidade desse de pertencer a uma determinada classe. Essa abordagem apresentou o problema de classificar objetos próximos de uma mesma classe como sendo um único objeto contíguo, para evitar esse problema, os autores utilizaram operações morfológicas sobre as imagens (como erosão, abertura, dilatação e máximo regional) sobre os *score maps*. Para garantir a consistência dos resultados, os autores realizaram validação cruzada separando 80% das imagens para treino e 20% das imagens para teste de forma aleatória e repetiram esse procedimento 30 vezes. Os seguintes resultados foram obtidos: revocação de 97,5%, 69,1%, 95,3% e 70,3%, precisão de 94,4%, 66,7%, 82,3% 82,2% para açaí, jací, paxiubão e espécie não identificada de palmeira, respectivamente, na abordagem convencional; e

revocação de 98,6%, 78,6%, 96,6% e 77,4%, precisão de 96,5%, 73,2%, 87,8% e 86,3% na abordagem proposta pelos autores utilizando operações morfológicas.

Casapia et al. [18] testaram o uso de pequenos UAV para identificar e contar espécies de palmeiras como buriti (*Mauritia flexuosa*), açaí (*Euterpe precatoria*), patauá (*Oenocarpus bataua*), entre outras, no nordeste do Peru por meio de segmentação semântica. Os autores avaliaram diferentes algoritmos de ML como k-Nearest Neighbours, Recursive Partitioning, Random Forest e Support Vector Machine Radial, obtendo melhores resultados com Random Forest, reportando 85% de acurácia média e 0,82 de índice *kappa*. Foi utilizado um UAV DJI Phantom 4 Pro e sua própria câmera para a captura das imagens, a uma altura de 60 e 90 m do solo, de modo a gerar 49 ortomosaicos. Apesar de não utilizar algoritmos de DL, o trabalho dos autores é completo, entrando em detalhes importantes sobre a obtenção de imagens, problemas que podem ocorrer durante a captura, seleção de *features*, entre outros.

Jintasuttisak, Edirisinghe e Elbattay [19] utilizaram imagens obtidas a partir de drone para detecção de tamareiras (*Phoenix dactylifera L.*) em plantações localizadas na região norte dos Emirados Árabes Unidos. O estudo analisa a viabilidade de diferentes versões da YOLOv5 (*small, medium, large* e *extra large*) frente a outras arquiteturas como a YOLOv3 [37], YOLOv4 [24] e SSD3000 [38]. O melhor resultado obtido foi com a versão *medium* da YOLOv5 [39], com mAP@0.50 de 92,34%, precisão de 91% e revocação de 92%. Os autores utilizaram um UAV senseFly eBee X com uma câmera senseFly S.O.D.A. para a captura das imagens, a uma altura de 122 m do solo, gerando 125 fotos. Essas fotos foram divididas em três conjuntos, para treino (60% equivalentes a 75 imagens), validação (20%, 25 imagens) e teste (20%, 25 imagens), com o conjunto de treino passando por transformações de *data augmentation*, aumentando seu quantitativo em 5 vezes.

Yarak et al. [20] propuseram um modelo para detecção e classificação da saúde de palmeiras em plantações na região sul da Tailândia utilizando imagens de UAV combinadas a técnicas de *deep learning*. Esse trabalho fez uso da arquitetura Faster R-CNN [40] utilizando elementos da Resnet-50 [36] e VGG-16 [34] para tanto, obtendo seus melhores resultados com a Resnet-50. O drone utilizado para captura das imagens foi o DJI Phantom 4 Pro utilizando sua própria câmera, os autores testaram diferentes altitudes de voo (100, 120, 140, 160, 180 e 200 m) e constataram melhor acurácia para detecção de palmeiras (independente da saúde dessas) a uma altura de 100 m do solo. Foram geradas 133 imagens de 2.000 $\times$ 2.000 px a partir dos ortomosaicos capturados durante o voo, com um total de 4172 caixas delimitadoras divididas em 3780 para treino (subdivididas em 3035 caixas para palmeiras saudáveis e 745 não saudáveis) e 392 (das quais 325 eram saudáveis e as demais não) para teste. Os melhores resultados obtidos com a Resnet-50 foram 95,09%, 92,07% e 86,96% de F1-score para a detecção de palmeiras (independente

da saúde), palmeiras saudáveis e doentes, respectivamente.

2.4 Aprendizagem de máquina e aprendizagem profunda

Aprendizagem de máquina é uma forma de se programar computadores para que esses aprendam por meio de dados [41]. Diferente da programação tradicional, onde é necessário codificar cada interação da máquina com uma dada entrada, algoritmos de aprendizagem de máquina utilizam os dados para calcular a resposta para um problema dado. Nesse contexto, são utilizados os conceitos de aprendizagem supervisionada, não supervisionada e por reforço. Na aprendizagem supervisionada, os dados recebem rótulos (do inglês, *labels*) previamente, de modo que um algoritmo possa aprender as características que definem um determinado rótulo. A abordagem não supervisionada dispensa o uso de rótulos e o próprio programa agrupa dados semelhantes por meio do procedimento definido. Por fim, algoritmos de aprendizagem por reforço baseiam-se na figura de um agente dentro de um ambiente, de modo que a interação entre agente e ambiente gera um reforço positivo ou negativo, e o agente deve aprender a melhor estratégia para conseguir o máximo de reforço positivo ao longo do tempo.

Atualmente, o termo aprendizagem de máquina é utilizado para se referir aos algoritmos como regressão linear, árvores de decisão, máquinas de vetor de suporte, entre outros. É possível enquadrar as redes neurais artificiais dentro desse grupo de algoritmos clássicos, entretanto, sua evolução levou ao que é comumente chamado de aprendizagem profunda. Redes neurais artificiais são modelos inspirados no funcionamento dos neurônios biológicos, codificados de maneira matemática. Um neurônio codificado nesse modelo matemático recebe os dados (sejam de entrada, sejam de outros neurônios; definidos como in_i na Equação 2.1) e os multiplica por um determinado peso ($w_{i,j}$), cujo resultado passa por uma função de ativação ($a(x)$) que gera uma saída (o_j), conforme a Equação 2.1. Essa saída pode ser o resultado final ou pode passar por outros neurônios que repetem essas operações. Nesse modelo, o aprendizado se dá pelo cálculo dos pesos aplicado a cada entrada, de modo que a saída se aproxime ao máximo da resposta esperada.

$$o_j = a\left(\sum_{i=0}^{n} w_{i,j} \cdot in_i\right) \tag{2.1}$$

A aprendizagem profunda caracteriza-se pelo uso de várias camadas de neurônios artificiais ligados uns aos outros em sequência. Essa área de ML se sobressai em problemas simples e intuitivos para humanos, porém, difíceis de serem descritos por meio de formalizações matemáticas, como o reconhecimento de fala ou de rostos de pessoas [22].

A arquitetura fundamental de redes neurais possui três camadas: de entrada, oculta e de saída. A camada de entrada é a responsável por receber os dados em sua dimensionalidade original, a camada oculta é a responsável por encontrar as correlações entre a entrada e a saída, podendo possuir múltiplos níveis de neurônios artificiais, ao passo que a de saída fornece o resultado do processamento dos dados. Uma rede neural artificial se torna profunda conforme o aumento do número de camadas de neurônios na camada oculta. De uma maneira geral, a rede se torna capaz de mapear situações mais complexas ao se utilizar mais camadas de neurônios artificiais, ao contrário de fazer o uso de um mesmo número de neurônios em menos camadas (*e.g.*, uma rede com 10 camadas ocultas, cada uma com 10 neurônios, é capaz de mapear um problema de complexidade maior do que uma segunda rede com 2 camadas ocultas e 50 neurônios cada).

A avaliação da aprendizagem de um modelo se dá pelo cálculo de uma função de perda (do inglês, *loss*), que afere o quanto a saída gerada difere da resposta esperada. A partir dessa computação, é possível atualizar os pesos da rede de modo a minimizar o valor da perda, o que caracteriza um problema de otimização. Dessa maneira, existem diversos algoritmos utilizados para esse problema, sendo os principais: o gradiente descendente estocástico (SGD, do inglês, *stochastic gradient descent*) e o Adam. O SGD é baseado no gradiente descendente [42] (descrito na Equação 2.2), onde são feitos ajustes de acordo com o sinal da derivada de uma função ($sign\left(f'(x)\right)$) e um parâmetro chamado de taxa de aprendizagem (ϵ). O gradiente descendente estocástico se difere por considerar apenas uma amostra selecionada aleatoriamente, ao invés de todo o conjunto de dados, para a atualização dos pesos, o que acelera o processo computacional.

$$f\left(x - \epsilon \cdot sign\left(f'(x)\right)\right) < f(x) \tag{2.2}$$

O gradiente descendente possui uma tendência a ficar preso em mínimos locais quando utilizado um valor baixo para o parâmetro ϵ. Por outro lado, um valor alto desse parâmetro pode distanciar o valor da perda de seu mínimo, levando o gradiente a divergir. Uma forma de agilizar a convergência e se aproximar do mínimo global é por meio da otimização por momento (do inglês, *momentum optimization*) [43], que leva em consideração o gradiente calculado nas iterações anteriores para atualizar a iteração corrente e acelerar a busca em direção ao mínimo global. O cálculo do momento e da atualização dos pesos seguindo essa estratégia é apresentado na Equação 2.3, onde m_i representa o momento para uma dada iteração i, β é um fator de decaimento (contido no intervalo de $[0, 1]$) que determina o quanto do momento da iteração anterior deve ser levado em consideração na iteração corrente, $J(x)$ representa a função de perda e θ, os pesos utilizados. Uma variação da otimização por momento se dá pelo gradiente acelerado de Nesterov (do inglês, *Nesterov Accelerated Gradient*) [44] que calcula o gradiente da função de perda em

um ponto a frente da posição local, na direção do momento. A Equação 2.4 apresenta a operação do gradiente acelerado de Nesterov.

$$\begin{aligned} m_i &= \beta m_{i-1} - \epsilon \cdot \nabla_{\theta_{i-1}} J(\theta_{i-1}) \\ \theta_i &= \theta_{i-1} + m_i \end{aligned} \tag{2.3}$$

$$\begin{aligned} m_i &= \beta m_{i-1} - \epsilon \cdot \nabla_{\theta_{i-1}} J(\theta_{i-1} + \beta m_{i-1}) \\ \theta_i &= \theta_{i-1} + m_i \end{aligned} \tag{2.4}$$

O otimizador Adam [45] utiliza conceitos da otimização por momento e do algoritmo RMSProp [46]. O RMSProp ajusta a taxa de aprendizagem com base na média móvel dos gradientes ao longo do tempo. O otimizador Adam ajusta a taxa de aprendizagem com base na média e variância dos gradientes. É um dos otimizadores mais populares, pois oferece velocidade de convergência com qualidade. O otimizador SGD associado ao gradiente acelerado de Nesterov oferece uma convergência mais lenta do que a do Adam, entretanto, pode oferecer maior qualidade nesse quesito [47].

Redes neurais convolucionais são um tipo de rede neural especializada no processamento de dados como imagens e séries temporais [22]. CNN têm em sua base filtros de convolução (também chamados de *kernels*, em inglês) que são utilizados para encontrar e extrair características relevantes dos dados. Apresenta-se a operação de convolução na Equação 2.5, onde $C(i, j)$ representa o mapa de ativação obtido a partir de uma função de convolução em duas dimensões (como em uma imagem), o operador $*$ denota a convolução de uma imagem I por um filtro K e i, j, m e n são índices utilizados para navegar pelo dado em suas dimensões. Uma desvantagem das operações de convolução é o aumento da dimensionalidade dos dados, exigindo uma quantidade maior de memória para armazená-los durante o treinamento da rede. Entretanto, é possível lidar com esse aumento na dimensionalidade pelo uso de camadas de *pooling* (do inglês, agregação).

$$C(i, j) = (I * K)(i, j) = \sum_m \sum_n I(m, n) \cdot K(i - m, j - m) \tag{2.5}$$

As camadas de *pooling* operam por meio de amostragem no dados, selecionando o maior valor ou a média de um subconjunto dos dados. Como não há o cálculo de um peso, essas camadas reduzem a dimensionalidade dos dados e, consequentemente, o uso de memória pela rede. CNN são amplamente utilizadas em tarefas de detecção de objetos [48], que dizem respeito a localizar e classificar determinadas instâncias em uma imagem. A tarefa de classificação fica ao encargo da CNN ao passo que a localização pode ser vista

como um problema de regressão onde, no caso da arquitetura YOLO, busca-se estimar as coordenadas normalizadas do centro do objeto e a largura e altura da BB, também normalizadas. A normalização, nesse caso, se dá pela divisão das coordenadas x e y do centro do objeto pela largura (l) e altura (h) da imagem, respectivamente, assim como a fração da largura (l_{bb}) e altura (h_{bb}) da caixa delimitadora pela largura e altura da imagem, na devida ordem, conforme a Equação 2.6. Dessa maneira, a *bounding box* calculada pode ser descrita como o seguinte vetor: $[x_{centro}, y_{centro}, l_{norm}, h_{norm}]$.

$$\begin{aligned} x_{centro} &= \frac{x}{l} & \qquad l_{norm} &= \frac{l_{bb}}{l} \\ y_{centro} &= \frac{y}{h} & \qquad h_{norm} &= \frac{h_{bb}}{h} \end{aligned} \tag{2.6}$$

2.5 Arquiteturas de *deep learning* da família YOLO

A YOLO é uma família de algoritmos de detecção de objetos em tempo real baseados em CNN. Abordagens anteriores [49, 50, 40] a dessa arquitetura consistiam em agregar duas redes diferentes: uma para propor localizações de objetos e outra para classificar esses objetos. Os autores da YOLO [51] propuseram a abordagem de unificar ambas as redes em um único algoritmo, de modo a realizar apenas uma passagem em cada imagem, propondo localizações e classificando ao mesmo tempo. Apesar de ter resultados de acurácia inferiores aos dos principais modelos da época, como a arquitetura R-CNN [50, 40], a YOLO era capaz de detectar objetos em tempo real com 63,4% de *mean average precision* (mAP) no conjunto de dados Pascal VOC 2007 [52]. Isto é, a YOLO era capaz de processar 45 imagens por segundo (FPS, do inglês, *frames per second*), podendo ser utilizada em vídeos (que são usualmente gravados em 24-30 FPS) por exemplo, ao passo que as arquiteturas da família R-CNN processavam de 0,5 a 18 imagens por segundo.

A segunda iteração da YOLO [53], nomeada YOLO9000, fez melhorias na arquitetura original, de modo que conseguiu superar modelos da família R-CNN na métrica de mAP (69,0-78,6%) sobre o conjunto de dados Pascal VOC 2007 com um processamento de imagens ainda mais rápido (91-40 FPS). Como é possível observar pelos intervalos dos resultados de mAP e FPS, há uma troca entre precisão e velocidade de processamento. A YOLOv3 [37] trouxe novas melhorias superando sua antecessora no conjunto de dados MS COCO [54] (não foram reportadas as métricas no conjunto Pascal VOC 2007). A YOLOv4 [24] aprimorou-se sobre sua iteração anterior, trazendo elementos dela e melhorando os resultados por meio do que os autores chamaram de "Bag of Freebies": métodos de *data augmentation* implementados na própria arquitetura da rede de modo a melhorar a performance de classificação a um custo de tempo mais alto ao longo do treinamento da

rede; e "Bag of Specials": métodos que melhoram a acurácia da rede a um baixo custo durante a execução. A YOLOv3 foi a última iteração publicada pelos autores originais, com a YOLOv4 sendo reconhecida por eles dado o número de contribuições de um de seus autores para o *framework* Darknet[4], que serviu de base para todas as versões da YOLO até então. Dessa maneira, existem novas iterações da família YOLO, com nomenclaturas diversas, entretanto, entre as mais significativas está a YOLOv4, que se estabeleceu como uma rede reconhecida na literatura.

A YOLOv9[25] é uma das iterações mais recentes da família YOLO. Seus autores partem do princípio do gargalo de informações (do inglês, *information bottleneck principle*) e constatam que a cada convolução da rede, há perda de informações. Uma forma de se evitar essa perda, é por meio da passagem do dado original para camadas mais profundas da rede, entretanto, isso gera um custo alto de processamento. Para minimizar esse custo, os autores fizeram uso de funções reversíveis, de modo que é possível obter o dado original por meio das propriedades desse tipo de função. Dessa maneira, a partir da implementação dos autores, é possível gerar gradientes confiáveis para atualizar os parâmetros da rede, utilizando função de perda de maneira mais adequada e evitando gerar falsas correlações em potencial no processo. A arquitetura da YOLOv9 foi construída tendo como base a YOLOv7 [55] e Dynamic YOLOv7 [56] para as versões "general" e "extended", respectivamente, obtendo resultados melhores sobre o conjunto de dados MS COCO que outras versões da família YOLO (como v5, v6, v7, v8, entre outras) e com um número menor de parâmetros.

2.6 Técnicas associadas à *machine learning*

No campo prático de aprendizagem de máquinas, são empregadas várias técnicas para aprimorar o desempenho dos modelos e assegurar resultados mais robustos e generalizáveis. Uma dessas técnicas é o *data augmentation* (do inglês, aumento de dados), que envolve a criação de novas amostras de dados a partir das existentes por meio de transformações como, no caso de imagens, rotação, espelhamento e distorção, aumentando assim a diversidade do conjunto de dados e ajudando a prevenir o *overfitting*. O *transfer learning* (do inglês, aprendizagem por transferência) é uma abordagem que permite a adaptação de modelos pré-treinados em grandes conjuntos de dados para tarefas específicas com menos dados, economizando tempo e recursos computacionais. Por fim, sistemas de *auto labeling* (do inglês, rotulagem automática) permitem a automatização do processo de anotação de dados, utilizando algoritmos para rotular novos dados com base em padrões aprendidos, o

[4] `https://github.com/AlexeyAB/darknet`

que é particularmente útil em grandes conjuntos de dados onde a rotulagem manual seria impraticável.

Data augmentation é uma técnica amplamente utilizada no contexto de *deep learning* e visão computacional [57, 58]. No caso de imagens, além das transformações mencionadas anteriormente, também são utilizadas manipulação de parâmetros como cor, contraste e brilho, apagamento aleatório, redimensionamento aleatório, fusão de imagens, entre outros. Ao manipular o contraste, por exemplo, o modelo pode se tornar mais tolerante a mudanças de iluminação, ao passo que mudanças na geometria da imagem (como rotação ou espelhamento) tornam o modelo mais tolerante a variações na posição [41]. O *random resizing* (do inglês, redimensionamento aleatório) é uma técnica de *data augmentation* que consiste no redimensionamento dos parâmetros de largura e altura da imagem por um fator escolhido aleatoriamente. Nesse caso, a imagem pode ser ampliada ou reduzida alterando características geométricas da imagem e detalhes finos de seus componentes.

Transfer learning consiste na tentativa de melhorar a performance de modelos de DL em um conjunto de dados por meio do treinamento prévio desses modelos em um outro *dataset* diferente, porém, relacionado [59]. Arquiteturas como a Faster R-CNN e YOLOv4 foram treinadas inicialmente no conjunto de dados ImageNet [60] e os pesos obtidos são utilizados no treinamento dessas redes sobre o *dataset* MS COCO, por exemplo. A técnica de *transfer learning* parte do princípio que as camadas de uma rede neural artificial mais próximas da entrada modelam características mais grosseiras (como segmentos de linhas de diferentes formas e orientações), ao passo que as camadas intermediárias combinam essas características em estruturas geométricas (como círculos e quadrados) e as camadas mais próximas da saída combinam as características de suas antecessoras em estruturas de nível mais alto, como rostos, árvores, objetos, etc. [41]. Dessa maneira, é possível reutilizar características das camadas mais profundas (mais próximas da camada de entrada) e treinar apenas as camadas mais superficiais, em um processo chamado de *fine-tuning* (do inglês, ajuste fino).

Como comentado anteriormente, modelos de *deep learning* requerem grandes quantidades de dados para conseguir gerar uma solução adequada e generalizável. Para além dos dados, modelos de aprendizagem supervisionada exigem a rotulagem desses para que possam operar. No contexto de detecção de objetos, sistemas de *auto labeling* auxiliam e aceleram a tarefa de rotulagem dentro do conjunto de dados. Técnicas comuns para implementar *auto labeling* incluem aprendizado supervisionado, onde um modelo é treinado em um conjunto de dados rotulado e, em seguida, usado para rotular novos dados, e aprendizado semi-supervisionado, que combina uma pequena quantidade de dados rotulados com uma grande quantidade de dados não rotulados para melhorar a precisão do modelo em relação a um modelo treinado apenas com dados rotulados [61].

Capítulo 3

Metodologia

Este capítulo aborda a metodologia seguida no trabalho. Comenta-se sobre o conjunto de dados utilizado, seu tratamento e o processo de anotação das imagens e, em seguida, descreve-se como foi realizado o treinamento e a avaliação das arquiteturas YOLOv4 e YOLOv9.

3.1 Conjunto de dados da Embrapa

As imagens utilizadas para compor os ortomosaicos foram coletadas no estado do Ceará em fevereiro de 2017, em duas regiões de aproximadamente 500 ha de área cada, nos municípios de Barbalha e Crato, fazendo fronteira com a Chapada do Araripe. Essa região é caracterizada pela ocorrência natural da macaúba-barriguda (*Acrocomia intumescens*) e babaçu (*Attalea speciosa*). O clima na região é classificado como `As` na classificação climática de Köppen, *i.e.*, tropical com verão seco [62] e a média de chuva anual na região é de 1.214 mm com desvio padrão de 353 mm, de acordo com registros da estação pluviométrica localizada no município de Barbalha no período de 1973 a 2010 (código 82784 da Organização Mundial de Meteorologia; instituto: Instituto Nacional de Meteorologia (INMET)), sendo que aproximadamente 85% do volume de chuva ocorre durante os meses de dezembro a abril. O tipo de solo predominante na região é o Argissolo Vermelho eutrófico (Acrisols no sistema FAO de classificação de solo; Ultisols na Taxonomia do Solo) [63]. A elevação média na região é de 648 m com desvio padrão de 48 m de acordo com o modelo digital de elevação derivado da missão AW3D30 do satélite Daichi [64].

Originalmente, essas imagens coletadas foram utilizadas para o treinamento de um modelo de detecção e contagem desenvolvido por pesquisadores da Embrapa[1]. Para a captura das fotos, foi utilizado um UAV DJI Phantom 4 equipado com uma câmera RGB de 12,1 megapixels de resolução e uma lente de 20 mm. O voo foi feito de modo que

[1]O trabalho não havia sido publicado até a entrega desta monografia.

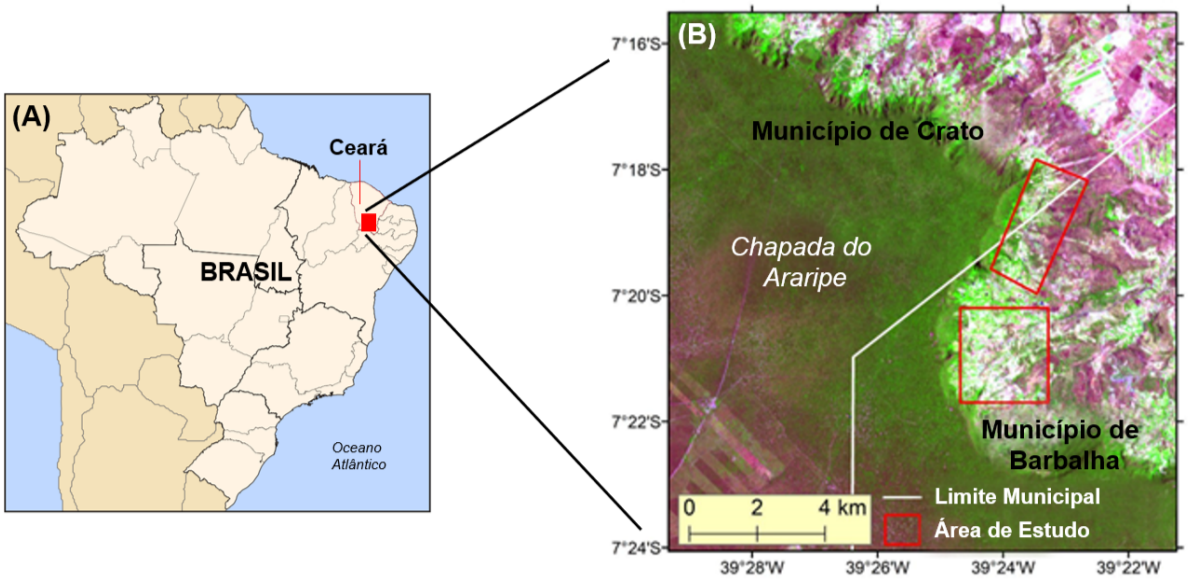

Figura 3.1: Locais de coleta das imagens no estado do Ceará.

as faixas cobertas pelas imagens possuem uma sobreposição de 60% tanto lateralmente quanto longitudinalmente. Os ortomosaicos gerados por essas imagens possuem distância amostral do solo (GSD, do inglês, *ground sampling distance*) de 3 a 20 cm.

3.2 Divisão dos ortomosaicos

A Embrapa cedeu um total de 14 ortomosaicos, sendo 10 com resolução de 20.000 × 20.000 px e os 4 restantes com 30.000 × 30.000 px. O primeiro passo foi dividi-los em imagens menores, optou-se pelo tamanho final de 2.000 × 2.000 px, pois, estima-se que a área útil dos ortomosaicos de 30.000 × 30.000 px seja maior, e o tamanho final escolhido gera um número inteiro de imagens, utilizando um *overlap* de 30% entre os cortes, ou seja, não seria necessário cortar ou utilizar alguma imagem fora do padrão de tamanho. A escolha da sobreposição de 30% é padrão na literatura para a construção de ortomosaicos por meio de *stitching* e, neste trabalho, tem o intuito de garantir que uma instância que seria cortada em uma divisão sem esse recurso possa ser observada pelo algoritmo, maximizando a quantidade de dados úteis. Por fim, imagens com mais de 90% de sua área preenchida pela cor preta foram descartadas, gerando um total de 2.032 arquivos.

Os ortomosaicos em questão são apresentados na Figura 3.2. Os nomes dos arquivos estão dispostos nessa Figura conforme foram passados ao autor. Originalmente, os ortomosaicos estão em formato TIFF e foram convertidos para a extensão `.jpg` após a

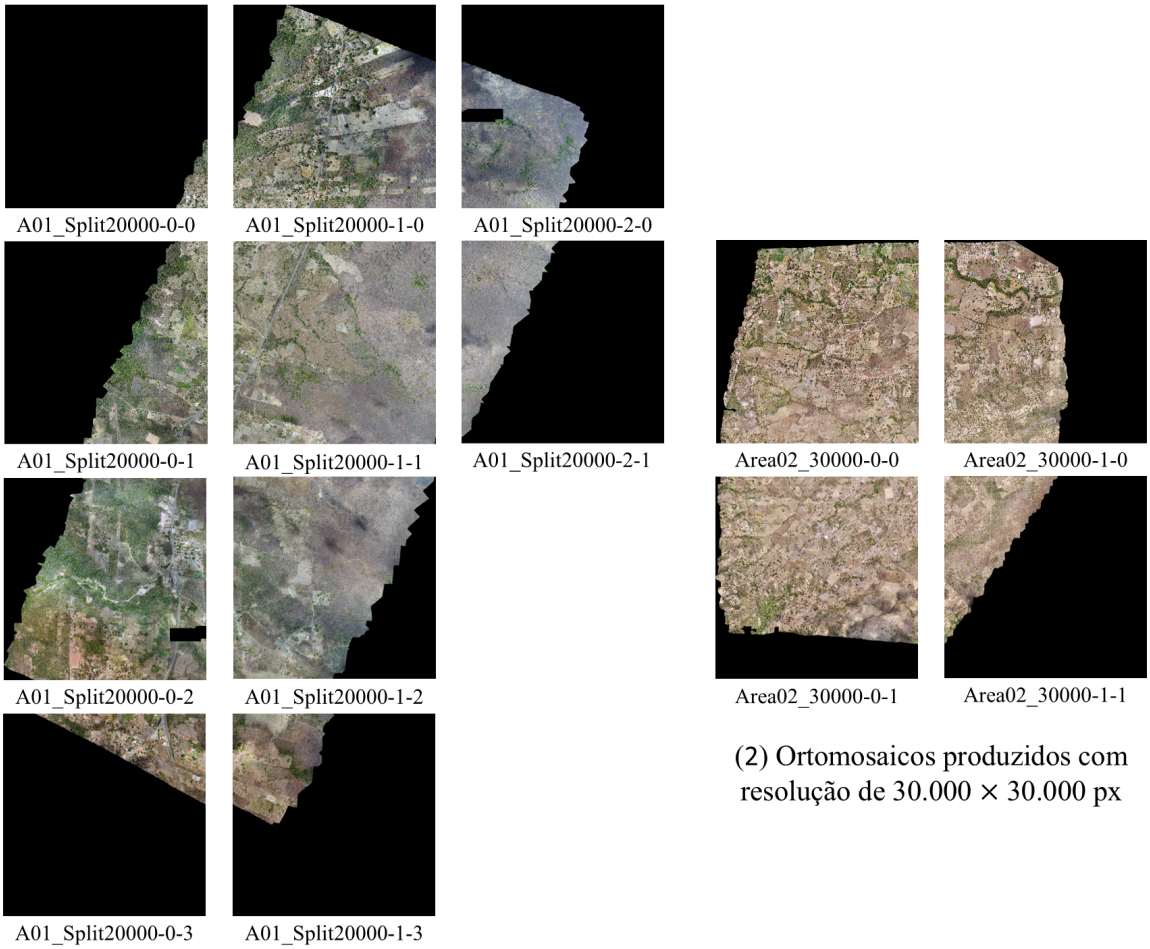

Figura 3.2: Ortomosaicos produzidos com imagens na divisa entre os municípios de Crato e Barbalha (1) e exclusivamente no município de Barbalha (2).

divisão. A escolha do formato JPEG se deu devido a esse ter um tempo de treinamento menor na YOLOv4 em relação à extensão `.png`[65].

3.3 Marcação das imagens

Com os ortomosaicos divididos em imagens de tamanhos iguais, o próximo passo foi a marcação das caixas delimitadoras. Para isso, utilizou-se a ferramenta Label Studio[2], por ser reconhecida no meio e possuir suporte para recursos de *auto labeling*. Esse último recurso é importante dado que o quantitativo de imagens neste trabalho foi anotado

[2] `https://labelstud.io/`

somente pelo autor. Inicialmente, rotulou-se 50 imagens sem nenhum tipo de auxílio e essas imagens foram utilizadas para o ajuste fino de uma rede YOLOv4 na configuração `yolov4-mish-416` com a camada de entrada reajustada para a dimensão de 480 × 480 px, *learning rate* de 10^{-5} e demais configurações mantidas conforme padrão da arquitetura. Com a rede treinada, foi feita a integração com a ferramenta de *auto labeling* do Label Studio, de modo que, todas as imagens rotuladas a partir desse ponto, passavam por uma classificação preliminar da rede treinada, e recebiam marcações que eram ajustadas ou removidas pelo autor que também marcava qualquer instância não reconhecida pela rede.

A integração de um modelo de DL com o Label Studio para produção do sistema de *auto labeling* se provou mais difícil do que a documentação [66] propunha. Foi preciso ir além do sistema sugerido (MMDetection[3]) e fazer a ligação entre a API (do inglês, *application programing interface*, ou interface de programação de aplicação) do Label Studio e um *script* em Python disponibilizado pelos autores da YOLOv4. Outro problema encontrado se deu com a conversão das dimensões das caixas delimitadoras. As imagens eram redimensionadas a 480 × 480 px para o modelo da YOLOv4 e anotadas em seu tamanho original de 2.000 × 2.000 px. Existe uma função do Label Studio para lidar com esse redimensionamento das *bounding boxes* para as diferentes resoluções utilizadas, a qual não funcionou corretamente, fazendo com que as caixas delimitadoras fossem registradas em coordenadas fora dos limites da imagem. A correção desse problema se deu por meio do desenvolvimento de uma solução em código, dispensando o uso da função em questão. Com o sistema integrado, foi dada sequência no processo de marcação das imagens.

Desse modo, após rotular mais 50 imagens foi feito um novo treinamento, resultando em uma atualização dos pesos e o procedimento foi sendo repetido. Os treinamentos subsequentes foram feitos com 200, 300 e 400 imagens rotuladas. As 100 primeiras imagens foram rotuladas sequencialmente, com o objetivo de facilitar o treinamento e compreensão das características das palmeiras observadas, sendo as 400 demais selecionadas aleatoriamente dentre o quantitativo disponível. Em todos os treinamentos, 10% das imagens eram reservadas para teste, de modo a avaliar e garantir que o modelo não sofria com *overfitting*. O conjunto de teste recebia novas imagens escolhidas aleatoriamente a cada novo treinamento, sempre mantendo as imagens anteriores, com o objetivo de garantir uma solução capaz de lidar com novos dados. A relação das imagens utilizadas e seus usos em cada uma das versões do modelo de *auto labeling* está disponível na Tabela C.1 do Apêndice C.

O treinamento do modelo de *auto labeling* se deu de maneira sequencial, com a primeira versão sendo um *fine-tuning* dos pesos pré-treinados sobre a base MS COCO e disponibilizados pelos autores da YOLOv4 e as iterações seguintes partindo dos pesos que

[3] `https://mmdetection.readthedocs.io/en/latest/`

obtiveram os melhores resultados na métrica mAP@0.50, utilizando uma estratégia análoga ao *early stopping*. O modelo da versão 1 foi treinado por 6.000 épocas, obtendo seus melhores resultados em 3.300 iterações. A segunda versão do modelo foi treinada com 100 imagens e obteve seu melhor resultado por volta da época 4.300 (*i.e.*, mil iterações após a primeira versão). A terceira versão obteve seu melhor resultado após 5.100 iterações, a quarta após 5.800 e a quinta obteve após 5.900. Os modelos da versão 2 em diante foram treinados por mais épocas além daquela onde obtiveram seus melhores resultados (em média, 4.000), mas os melhores resultados eram obtidos após 1.000 iterações. O número maior de épocas de treinamento foi utilizado para averiguar se o modelo não estava preso em um mínimo local. O treinamento desse modelo foi feito no computador pessoal do autor, com as seguintes especificações: CPU AMD Ryzen 7 5800X3D, 16 GB de memória RAM e placa de vídeo EVGA GeForce GTX 1070 FTW com 8 GB de memória VRAM.

Figura 3.3: Exemplos de distorções encontradas na imagem `A01_Split20000-0-1_patch__088.jpg`. A foto a esquerda representa a imagem `A01_Split20000-0-1_patch_088.jpg`, reduzida a 15% de seu tamanho original. As fotos marcadas com os números 1, 2 e 3 são recortes da imagem original reduzidas a 60%, no caso dos números 1 e 2, e 50% no caso do 3. A figura foi reduzida novamente para inserção neste trabalho

Ao longo do processo de marcação das imagens, dada a falta de conhecimento a nível de especialista para identificação das espécies, foram utilizadas algumas imagens disponibilizadas pela Embrapa com exemplares de babaçu e macaúba e suas respectivas caixas delimitadoras para orientar esse procedimento. Algumas imagens apresentavam um tipo de distorção exemplificado na Figura 3.3. Inicialmente, pensou-se que poderiam ser artefatos causados durante a divisão das imagens, porém, essas distorções foram observadas também nos ortomosaicos. Entretanto, após a leitura do trabalho de Casapia et al. [18], se constatou que esses defeitos nas imagens aparentam ter sido causados durante o processo de *stitching*, especificamente, devido ao vento ao longo do processo de captura das

Tabela 3.1: Quantidade de marcações por classe.

Classe	Quantidade
Babaçu	1.760
Macaúba	1.697
Total de marcações	3.457
Imagens sem marcações	54

imagens, que pode ter modificado a posição das palmeiras entre uma captura e as subsequentes. No início do processo de marcação, tomou-se uma abordagem de marcar todo tipo de palmeira que se assemelhasse a uma das espécies desejadas, independentemente do nível de distorção nos espécimes, entretanto, ao final do processo, foram marcados apenas exemplares que pudessem permitir algum grau de certeza quanto a sua marcação. Essa decisão foi tomada dada a falta de conhecimento específico sobre a identificação das espécies, com o intuito de garantir que o modelo fosse treinado sobre dados com maior grau de corretude e, dessa maneira, todas as imagens foram revisadas para garantir que suas marcações estivessem padronizadas.

3.4 Conjunto de dados utilizado

No período de agosto a setembro de 2023, 500 imagens foram marcadas gerando um total de 3.457 caixas delimitadoras. Do quantitativo de caixas delimitadoras, 1.760 são relativas à espécie de babaçu e 1.697, de macaúba, sendo que 54 imagens não possuem nenhum tipo de marcação. Imagens sem marcação também são importantes para que o modelo aprenda a diferenciar as palmeiras do plano de fundo das fotografias. Essas informações estão sintetizadas na Tabela 3.1.

Para analisar a distribuição do número de marcações por imagem, foram confeccionados os histogramas apresentados nas Figuras 3.4 a 3.6. É possível observar que a grande maioria das imagens possui entre 1 a 9 marcações, entretanto, algumas imagens possuem 20 ou mais marcações. Para ambas espécies de palmeiras, a maioria das imagens possuem entre 1 a 4 marcações, com o número diminuindo no intervalo de 5 a 9 e a tendência de redução se mantém nos intervalos subsequentes. Existem algumas poucas imagens de babaçu com 25 a 30 marcações, mas nenhuma supera o valor de 30 marcações, como é possível observar na Figura 3.5. Por outro lado, algumas imagens superam o quantitativo de 30 marcações de macaúba, conforme apresentado na Figura 3.6.

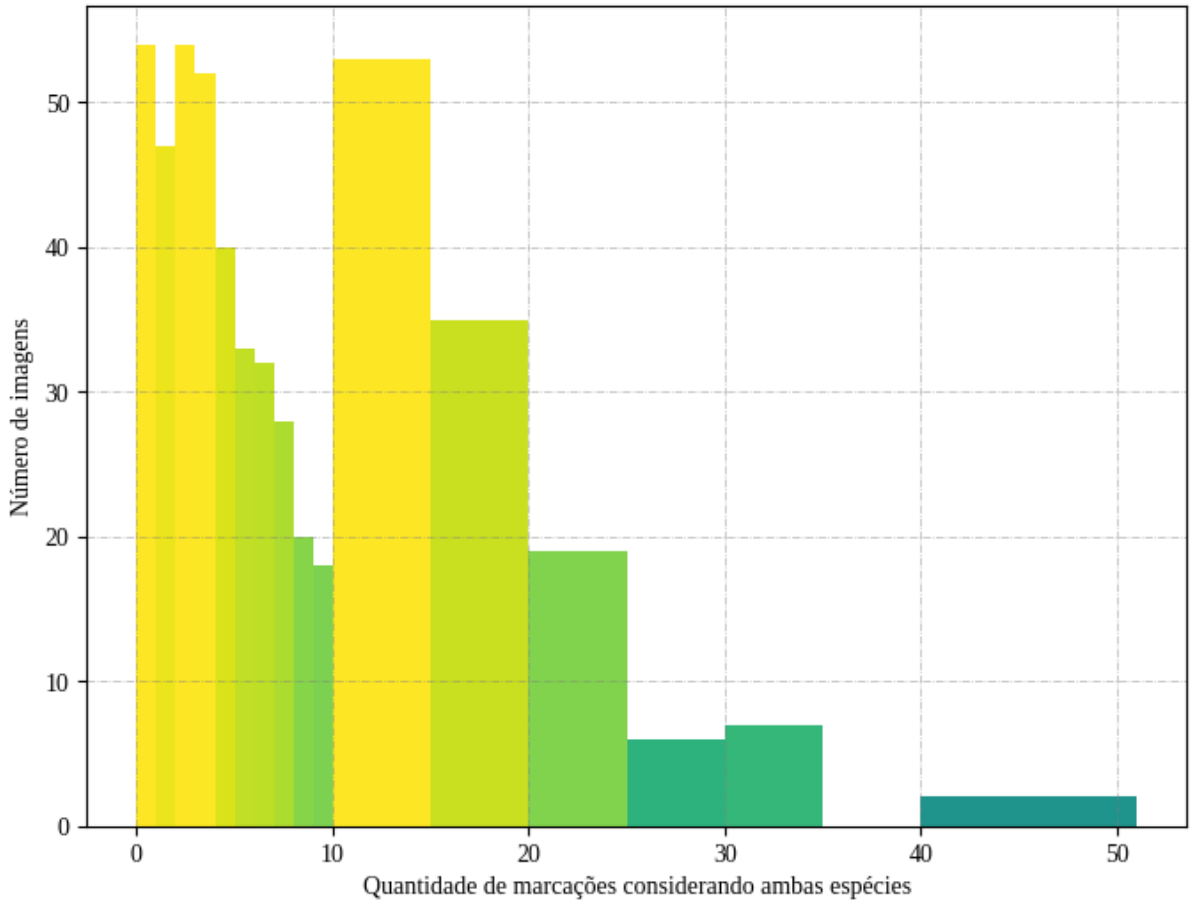

Figura 3.4: Histograma do número de marcações por imagens. Entre os valores de 1 a 10, o intervalo é unitário e de 10 em diante o intervalo é de 5.

3.5 Modelos utilizando a arquitetura YOLOv4

Dois modelos distintos foram treinados utilizando a arquitetura YOLOv4. O primeiro se baseia na ideia de um modelo de produção, onde todas as imagens foram utilizadas para treinamento a partir dos pesos pré-treinados sobre a base MS COCO. Como os resultados desse modelo estavam com uma performance abaixo do esperado, foi treinada uma segunda rede utilizando os pesos do modelo de *auto labeling*. Nesse último, os resultados foram melhores, mas abaixo do que foi obtido no modelo original de *auto labeling*. Ambas redes foram treinadas em um computador do Laboratório de Imagens, Sinais e Acústica vinculado ao Departamento de Ciência da Computação da Universidade de Brasília (LISA) com as seguintes especificações: CPU Intel Core i7-8700, 64 GB de memória RAM e duas placas de vídeo Nvidia GeForce RTX 2080 SUPER com 8 GB de memória VRAM, cada.

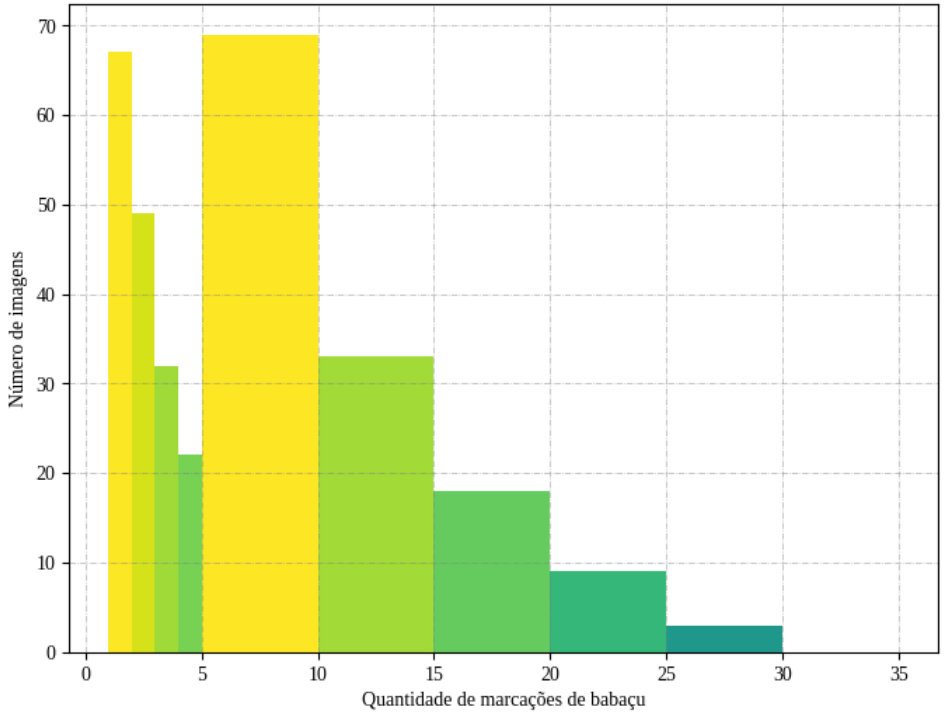

Figura 3.5: Histograma do número de marcações de babaçu por imagens. Entre os valores de 1 a 5, o intervalo é unitário e de 5 em diante, 5.

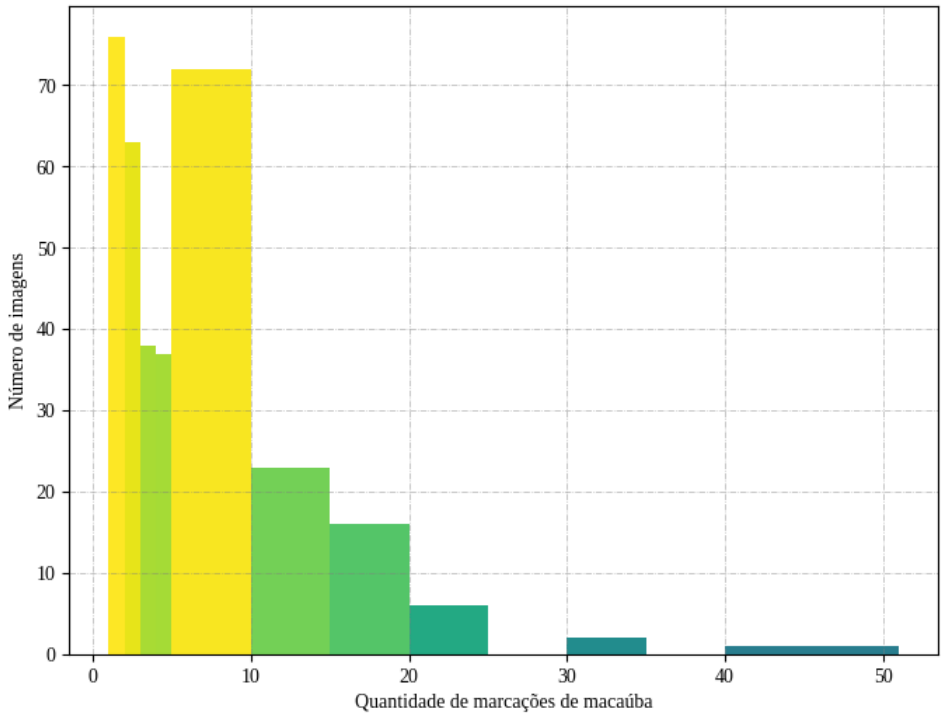

Figura 3.6: Histograma do número de marcações de macaúba por imagens. Entre os valores de 1 a 5, o intervalo é unitário e de 5 em diante, 5.

3.5.1 Modelo de produção utilizando os pesos pré-treinados sobre a base MS COCO

Em posse das imagens marcadas, foi feita uma nova seleção aleatória de imagens de treino e validação, na proporção de 90% e 10% respectivamente, para uma verificação inicial. A relação de imagens e seus usos é apresentada na Tabela C.1 do Apêndice C. A configuração da rede foi mantida na `yolov4-mish-416` (que possui em torno de 27 milhões de parâmetros) devido a limitação de memória VRAM das GPUs disponíveis, uma vez que, os autores da YOLOv4 sugerem um treinamento inicial por 1.000 épocas utilizando uma única placa gráfica antes de se usar múltiplas GPUs para o treinamento. Desse modo, foram realizados novos treinamentos e testes utilizando o quantitativo total de imagens disponíveis.

O primeiro teste realizado foi um novo *fine-tuning* dos pesos disponibilizados pelos autores da YOLOv4 previamente treinados sobre a base de imagens MS COCO. Foram mantidas as mesmas configurações utilizadas no modelo de auto labeling para fins comparativos. Como os resultados foram abaixo do esperado, realizou-se uma nova sessão de treinamento modificando o hiperparâmetro de taxa de aprendizado (LR, do inglês, *learning rate*) para 10^{-4}, resultando em ganhos de performance nas métricas avaliadas. Um terceiro teste foi conduzido aumentando a LR para 10^{-3}, entretanto, o gradiente ficou instável e seu valor aumentou significativamente.

Os testes seguintes visaram o aumento da resolução na camada de entrada da rede. Para tanto, os valores de altura e largura da imagem devem ser múltiplos de 32 (requisito da arquitetura). O quarto teste foi feito com as dimensões de 800 $\times$ 800 px, a maior possível dada a limitação de memória das placas gráficas. Para tanto, foi desabilitada a função contida na rede de redimensionamento aleatório das imagens. Em seguida, para testar os efeitos do redimensionamento aleatório, foram feitos testes nas dimensões de 640 $\times$ 640 px e 704 $\times$ 704 px, sendo a última a maior resolução possível dado os fatores de redimensionamento.

Após os testes sobre o aumento da resolução, também foram feitos testes modificando o hiperparâmetro de momento que, por padrão, está definido com o valor de 0,949 e mantendo a LR em 10^{-4}. Os valores de momento testados foram 0,975, 0,962 e 0,960. O primeiro valor testado não resultou em um ganho significativo e houve instabilidade no gradiente. O segundo valor obteve ganhos em mAP@0.50 durante o treinamento, mas esses resultados não se mantiveram sobre o conjunto de validação. Por fim, o último valor testado também não gerou ganhos na métrica avaliada. Dessa maneira, optou-se por manter o valor padrão de momento como melhor opção para treinamentos baseados nos pesos pré-treinados sobre a base MS COCO.

3.5.2 Modelo utilizando os pesos pré-treinados do sistema de *auto labeling*

O modelo treinado com os pesos ajustados para o sistema de *auto labeling* utilizou as mesmas imagens de treinamento que o sistema em questão acrescidas das últimas 100 marcadas, conforme a Tabela C.1 no Apêndice C. Inicialmente, se tentou fazer um *fine-tuning* a partir dos melhores pesos da Versão 5 (conforme a Tabela 4.1) mantendo-se as configurações utilizadas no modelos de *auto labeling*, o que não foi bem sucedido devido a instabilidade no gradiente. A segunda abordagem envolveu o uso da *flag* `-clear` para iniciar o treinamento. Essa *flag* reinicializa o treinamento, dessa maneira, é possível ter controle direto sobre a taxa de aprendizado inicial, momento e demais hiperparâmetros. Assim, reduziu-se a LR para 10^{-7} e o momento para $0,925$ e manteve-se a resolução da camada de entrada em 480×480 px. Desse modo, foi possível alcançar resultados semelhantes aos obtidos pelo modelo de produção da YOLOv4 com a resolução da camada de entrada definida para 704×704 px.

Como esse resultado positivo, aumentou-se a resolução da camada de entrada para 704×704 px e a LR para 10^{-6}, mantendo o valor de momento, e os resultados subiram 5 pontos percentuais. Em seguida, manteve-se as dimensões da camada de entrada e foram feitos testes modificando a LR para 10^{-5} e 10^{-4}, assim como o momento para 0,937. O melhor resultado obtido foi com os hiperparâmetros de LR igual a 10^{-4} e momento igual a 0,937. Desse modo, foi possível obter um ganho de quase 7% de mAP@0.50 utilizando os pesos treinados previamente para o sistema de *auto labeling*.

Por fim, foi feita a validação cruzada *k-fold* com os parâmetros que obtiveram o melhor resultado (resolução da camada de entrada igual a 704×704 px, LR igual a 10^{-4} e momento igual a 0,937). Como o modelo e seus pesos calculados já haviam observado 360 imagens durante seu treinamento, optou-se por fazer 5 divisões do conjunto de 140 (100 imagens marcadas posteriormente ao último treinamento do sistema de *auto labeling* e 40 utilizadas para teste). A relação das imagens utilizadas e seus respectivos conjuntos em cada *fold* está disponível na Tabela C.2 no Apêndice C. Assim, a validação cruzada em 5 *folds* foi feita da seguinte maneira: o modelo foi treinado por outras 5 vezes, a partir dos melhores pesos da Versão 5 do sistema de *auto labeling*, utilizando 472 imagens para treinamento e 28 para teste. Dessa forma, pretende-se avaliar os resultados e a consistência do modelo em 5 conjuntos de treinamento e teste diferentes.

3.6 Modelo utilizando a arquitetura YOLOv9

O treinamento do modelo com a YOLOv9 utilizou o mesmo conjunto de imagens para treino e teste utilizado pelo modelo de produção utilizando os pesos pré-treinados sobre a base MS COCO da YOLOv4. Esse conjunto de imagens foi escolhido para manter o padrão de comparação com essa versão do modelo da YOLOv4, pois ainda não havia sido feito nenhum tipo de treinamento com o modelo que utilizou os pesos pré-treinados do sistema de *auto labeling*. Foi utilizada a implementação da Ultralytics[4] da arquitetura da YOLOv9 (na configuração `YOLOv9e` que possui em torno de 58 milhões de parâmetros) e o mesmo computador utilizado pelos modelos de produção da YOLOv4. Inicialmente, foram utilizados os parâmetros padrão da rede para se realizar uma familiarização com sua estrutura de comandos.

A implementação da Ultralytics possui um sistema automatizado para a escolha do melhor otimizador para a rede, assim como a melhor taxa de aprendizado inicial. Como foram utilizadas as configurações padrão, esse sistema escolheu o otimizador AdamW com a taxa de aprendizado igual a $7,14 \cdot 10^{-4}$. Esse modelo foi treinado por 100 épocas e obteve o melhor resultado de mAP@0.50 dentre os modelos utilizando a arquitetura YOLOv9. Como esse treinamento foi o primeiro a ser feito, foram exploradas diferentes configurações para averiguar de maneira melhor o resultado obtido.

O segundo treinamento ocorreu por mais 100 épocas a partir do primeiro, para verificar se o modelo não estava preso em um mínimo local, entretanto, esse não era o caso. Em seguida, avaliou-se o uso do otimizador SGD (do inglês, *stochastic gradient descent*) com o gradiente acelerado de Nesterov, pois ambos otimizadores são complementares: o Adam converge rapidamente e com qualidade satisfatória, porém pode ter problemas em alguns conjuntos de dados, uma adversidade que não ocorre com o SGD associado ao gradiente de Nesterov [47]. Porém, o otimizador AdamW se sobressaiu no conjunto de dados deste trabalho. Também foi feito um teste reduzindo a taxa de aprendizado do otimizador AdamW para $1 \cdot 10^{-4}$ e treinando por 200 épocas, mas os resultados não foram satisfatórios. Outro otimizador testado foi o Adam no lugar do AdamW, entretanto, isso não levou a um aumento na métrica avaliada. Por fim, como ainda havia memória disponível nas placas de vídeo utilizadas, aumentou-se as dimensões da camada de entrada da rede, o tamanho padrão é 640 × 640 px e foram testadas as resoluções de 672 × 672 px e 704 × 704 px, porém, essa modificação não gerou melhoria nos resultados.

Dessa maneira, optou-se pelo uso do sistema automatizado disponibilizado pelo *framework* para os testes finais utilizando a validação cruzada *k-fold*. No caso da YOLOv9, o conjunto de dados foi dividido em 10 subconjuntos. Cada subconjunto continha 50

[4] `https://docs.ultralytics.com/models/yolov9/`

imagens, de modo que foram utilizadas 450 para treino e 50 para teste em cada iteração. Essa escolha se deu pois os pesos pré-treinados utilizados foram ajustados para o *dataset* MS COCO e não tiveram contato prévio com o conjunto de dados utilizado neste trabalho. Assim, a validação cruzada *k-fold* pode ser executada de maneira mais usual. A relação das imagens utilizadas e seus respectivos conjuntos em cada *fold* é apresentada na Tabela C.3 no Apêndice C.

3.7 Avaliação dos modelos

Para a avaliação inicial dos modelos, foram utilizadas as métricas de *mean average precision*, precisão, revocação e F1-score. Precisão é uma métrica que avalia a acurácia das predições positivas feitas por um modelo. A revocação mede a capacidade do modelo em capturar todos os objetos relevantes em uma imagem. A F1-score sumariza as métricas de precisão e revocação em uma, por meio da média harmônica delas. A métrica mAP é utilizada na avaliação de modelos de detecção de objetos e calcula a média das precisões obtidas em diferentes níveis de revocação para as diferentes classes.

Para melhor definir cada uma das métricas, é necessário explorar suas variáveis. Começando por um problema inerente a detecção de objetos, é preciso definir quando o modelo acerta uma detecção. Para isso, utiliza-se do conceito de interseção sobre a união (IoU, do inglês, *intersection over union*), na qual a interseção entre as áreas da caixa delimitadora proposta pelo modelo (A) e de *ground truth* (B) é dividida pela união das áreas dessas mesmas caixas delimitadoras, conforme a Equação 3.1. Idealmente, esse parâmetro se aproxima de 1 para um acerto total, entretanto, também são utilizados outros limiares, como $0,05$, $0,50$, $0,90$, $0,95$, etc.

$$IoU = \frac{\text{Área da interseção}}{\text{Área da união}} = \frac{|A \cap B|}{|A \cup B|} \tag{3.1}$$

Outra definição importante se dá pela terminologia de verdadeiro positivo (TP, do inglês, *true positive*), falso positivo (FP), verdadeiro negativo (TN, do inglês *true negative*) e falso negativo (FN). Em problemas multiclasse (como é o caso deste trabalho, onde há as classes de babaçu, macaúba e plano de fundo), essas variáveis são analisadas da seguinte maneira:

- TP_i: instâncias classificadas corretamente como pertencentes a uma determinada classe i;
- FP_i: instâncias classificadas incorretamente como pertencentes uma determinada classe i;

- TN_i: instâncias classificadas corretamente como não pertencentes a uma determinada classe i; e
- FN_i: instâncias de uma determinada classe i classificadas incorretamente como de outra classe.

Assim, calcula-se as métricas de precisão e revocação para um determinado limiar de IoU. A precisão de uma classe i ($Prec_i$), calculada como na Equação 3.2, avalia, entre os elementos classificados como pertencentes a uma determinada classe ($TP_i + FP_i$), qual a proporção rotulada corretamente.

$$Prec_i = \frac{TP_i}{TP_i + FP_i} \tag{3.2}$$

A métrica de revocação para uma classe i (Rec_i) é calculada segundo a Equação 3.3 e avalia, dentre o total de elementos que deveriam ser classificados como pertencentes a determinada classe ($TP_i + FN_i$), qual a proporção rotulada corretamente.

$$Rec_i = \frac{TP_i}{TP_i + FN_i} \tag{3.3}$$

Como comentado anteriormente, a métrica F1-Score de uma determinada classe i ($F1_i$), calculada conforme a Equação 3.4, representa a média harmônica entre as métricas de precisão e revocação.

$$F1_i = \frac{2 \cdot (Prec_i \cdot Rec_i)}{(Prec_i + Rec_i)} \tag{3.4}$$

Uma outra forma de sintetizar uma métrica é por meio da média aritmética ($\bar{x}$) dessa, que é calculada somando seus valores para todas as i diferentes classes e dividindo esse somatório pelo número total de classes (n), como na Equação 3.5.

$$\bar{x} = \frac{1}{n} \cdot \sum_{i=1}^{n} \text{Métrica}_i \tag{3.5}$$

Ao se observar as métricas de precisão e revocação, é possível perceber que há uma relação de troca entre elas. Um modelo otimizado para maior precisão tende a classificar menos amostras como pertencentes a uma determinada classe, levando a um aumento de falsos negativos e, consequentemente, a uma perda na revocação. Por outro lado, um modelo otimizado para revocação, tende a classificar mais amostras de maneira errônea, gerando mais falsos positivos, o que reduz a precisão. Uma maneira de se avaliar essa troca é por meio da *average precision*, que mede a precisão de um modelo ao longo de diferentes níveis de revocação. A *mean average precision* diz respeito a média da *average precision* em diferentes níveis de IoU. Uma métrica comumente utilizada em modelos de

detecção de objetos é a mAP@0.50, *i.e.*, a *mean average precision* para um valor de IoU de no mínimo 50%.

Para a avaliação final dos modelos, foi utilizada a técnica de *k-fold cross validation* (do inglês, validação cruzada *k-fold*). O procedimento consiste em dividir o conjunto de dados em k subconjuntos aproximadamente iguais. Em cada uma das k iterações, um subconjunto diferente é utilizado para validação e os $k - 1$ demais para treinamento. Isso garante uma estimativa mais confiável da performance do modelo uma vez que há maior variabilidade nos dados de treino e validação, permitindo uma melhor avaliação da capacidade de generalização do modelo uma vez que as métricas avaliadas são coletadas em cada uma das iterações.

Capítulo 4

Resultados experimentais

Neste capítulo, são apresentados os resultados obtidos pelos modelos treinados e a discussão desses resultados.

4.1 Modelo de *auto labeling*

Os resultados do modelo *de auto labeling* são apresentados aqui, inicialmente, para fins de registro, uma vez que o foco do trabalho se dá nos modelos descritos nas Seções seguintes. As métricas do treinamento desse modelo sobre o conjunto de testes, assim como a relação entre as versões e a quantidade de imagens estão sintetizadas na Tabela 4.1. Os gráficos de treinamento do modelo são apresentados na Seção A.1 do Apêndice A. Como comentado anteriormente, os treinamentos das versões 1 e 2 foram feitos utilizando imagens em sequência, *i.e.*, utilizando fotos dos ortomosaicos `A01_Split20000-0-0` e `A01_Split20000-0-1`, seguindo sequencialmente da direita para a esquerda, de cima para baixo, com uma sobreposição de 30% entre as laterais das imagens. Naturalmente, isso levou a rede a um problema de sobreajuste (do inglês, *overfitting*), onde as espécies eram identificadas apenas quando se encontravam sobre um plano de fundo verde, sendo que em algumas imagens as espécies se encontravam sobre planos de fundo com uma coloração de tom amarelo queimado. Esse problema foi corrigido entre as versões 3 e 4.

A partir da versão 4, notou-se que o modelo marcava quase toda palmeira encontrada nas imagens como sendo da espécie babaçu. Isso foi um motivador para a mudança de estratégia de marcação, saindo da abordagem de marcar toda palmeira que se assemelhasse a uma das espécies para a conduta de marcar apenas exemplares que pudessem permitir um grau de confiança maior sobre a sua marcação. É possível observar essa mudança na queda dos resultados obtidos entre a versão 4 e 5, uma vez que as imagens marcadas anteriormente ainda não haviam passado por revisão.

Tabela 4.1: Resultados do treinamento da ferramenta de *auto labeling*.

Versão	Total de imagens	Métricas			
		mAP@0.50	Precisão	Revocação	F1-score
1	50	78,09%	0,67	0,80	0,73
2	100	81,78%	**0,80**	0,74	0,77
3	200	82,17%	0,72	0,74	0,73
4	300	**86,82%**	0,74	0,83	**0,78**
5	400	84,04%	0,65	**0,85**	0,74

Os resultados iniciais foram surpreendentes, com apenas 45 imagens de treinamento foi possível alcançar 78% de mAP@0.50. Entretanto, neste ponto, o conjunto de dados era simples e homogêneo por escolha. Não houve um ganho significativo entre a versão 2 e 3, mas isso reflete a robustez da arquitetura quando confrontada com novos exemplares para treinamento, afinal, neste ponto, as imagens deixaram de ser sequenciais para serem escolhidas aleatoriamente pela ferramenta do Label Studio, de modo que haviam 100 imagens sequenciais e 100 imagens escolhidas aleatoriamente. Houve um salto na performance entre a versão 3 e 4, provavelmente, devido a inclusão de novas imagens em condições diferentes, permitindo que a rede fosse capaz de identificar melhor as características das palmeiras de babaçu e macaúba.

De uma maneira geral, o modelo de *auto labeling* foi de grande ajuda para a marcação das imagens. Sempre era necessário realizar ajustes das marcações, assim como marcar alguma instância ignorada pelo modelo, mas ele realizava a maior parte do trabalho e seu comportamento estava dentro do esperado. Houve algumas dificuldades durante a implementação do sistema, como comentado anteriormente, porém, elas foram superadas. Assim, considera-se esse modelo como adequado para o que lhe era proposto.

4.2 Modelos utilizando YOLOv4

4.2.1 Modelo de produção utilizando pesos pré-treinados sobre a base de dados MS COCO

A principal diferença entre este modelo e o anterior é que se considera este como um modelo de produção. Como comentado anteriormente, este modelo teve acesso a todas as 500 imagens disponíveis para os processos de treinamento e teste desde o início. A primeira versão do modelo foi um teste comparativo com a rede treinada para o sistema de *auto labeling*, foram utilizadas as mesmas configurações e esse modelo foi treinado por 6.000 épocas. Como os resultados se mostraram insuficientes (com o melhor resul-

tado de mAP@0.50 alcançando 59,69%), o modelo foi treinado por mais 4.000 iterações, totalizando 10.000 épocas, resultando em um mAP@0.50 melhorado de 63,43%.

Dessa maneira, para as versões seguintes alterou-se os hiperparâmetros da rede, iniciando pelo número total de épocas, que foi elevado a 10.000. Nas versões 2 e 3, a taxa de aprendizagem foi incrementada aos valores de 10^{-4} e a 10^{-3}, respectivamente. O valor de 10^{-4} se mostrou o melhor para a tarefa, com 67,43% de mAP@0.50, ao passo que o modelo com LR igual a 10^{-3} sofreu instabilidade do gradiente e obteve melhor resultado de 14,38% de mAP@0.50. As métricas obtidas sobre o conjunto de teste e parâmetros e hiperparâmetros modificados na rede estão sintetizadas na Tabela 4.2 e os gráficos de treinamento são apresentados na Seção A.2 do Apêndice A.

Tabela 4.2: Resultados de treinamento do modelos de produção YOLOv4 (utilizando pesos pré-treinados sobre a base de dados MS COCO) sobre seu conjunto de validação

Versão	Resolução	LR	Momento	Métricas			
				mAP@0.50	Precisão	Revocação	F1-score
1	480 × 480	10^{-5}	0,949	63,43%	0,52	0,64	0,58
2	480 × 480	10^{-4}	0,949	67,43%	0,53	0,73	0,62
3	480 × 480	10^{-3}	0,949	14,38%	0,30	0,14	0,20
4	800 × 800	10^{-4}	0,949	**70,88%**	0,48	0,87	0,62
5	640 × 640	10^{-4}	0,949	69,53%	**0,55**	0,77	0,64
6	704 × 704	10^{-4}	0,949	69,92%	0,50	0,86	0,64
7	704 × 704	10^{-4}	0,975	68,85%	0,54	0,83	**0,65**
8	704 × 704	10^{-4}	0,962	68,40%	0,50	**0,88**	0,63
9	704 × 704	10^{-4}	0,960	69,56%	0,52	0,84	0,64

Estabelecido o melhor valor para a taxa de aprendizagem, as versões subsequentes exploraram o efeito do aumento da resolução da imagem na camada de entrada da rede sobre os resultados. Na versão 3 testou-se a maior resolução possível para a imagem dadas as limitações de memória VRAM das placas gráficas utilizadas: 800 × 800 px. Acima dessa resolução, não era possível treinar um modelo YOLOv4 na configuração `yolov4-mish-416`. Ao se utilizar essas dimensões, obteve-se mAP@0.50 de 70,88%. Esse resultado serviu de base para explorar o efeito do redimensionamento aleatório na performance do modelo, uma vez que foi preciso desabilitar essa funcionalidade na rede para executar o treinamento.

As versões 5 e 6 testaram as dimensões de 640 × 640 px e 704 × 704 px, respectivamente, utilizando o *random resizing*. A última resolução foi a maior encontrada que permitia o uso dessa funcionalidade durante o treinamento. As versões 5 e 6 alcançaram resultados de 69,53% e 69,92% de mAP@0.50, respectivamente. Apresenta-se o gráfico de treinamento da versão 6 na Figura 4.1. Naturalmente, o treinamento dessas versões se deu em menor tempo do que o da versão 4 que totalizou 11,43 horas (h), com as ver-

sões 5 e 6 totalizando 9,08 e 10,71 h, respectivamente. Acredita-se que o uso do *random resizing* é vantajoso em comparação ao aumento da resolução, pois o primeiro resulta em maior variabilidade nos dados de treinamento, ao custo de uma leve perda na métrica mAP@0.50, mas um leve ganho em F1-score.

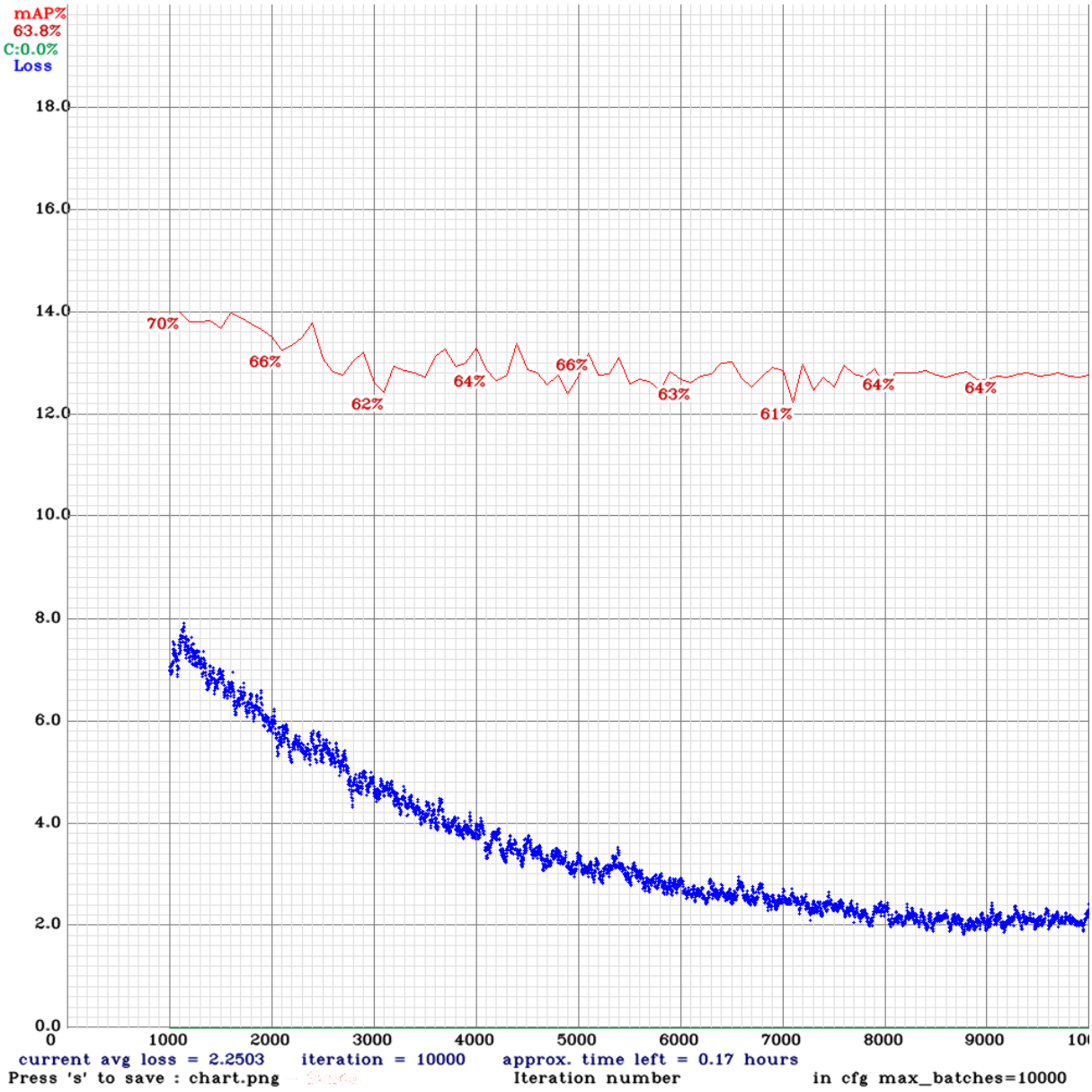

Figura 4.1: Gráfico da perda (em azul) e mAP@0.50 (em vermelho) em função das épocas relativo ao treinamento da versão 6 do modelo de produção da YOLOv4 utilizando pesos pré-treinados sobre a base de dados MS COCO.

As versões 7, 8 e 9 testaram diferentes valores para o parâmetro de momento. O valor de 0,975 da versão 7 gerou instabilidade no gradiente e não apresentou nenhuma vantagem. Durante o treinamento da versão 8, o modelo atingiu 70,65% de mAP@0.50,

Tabela 4.3: Resultados de treinamento dos modelos de produção YOLOv4 (utilizando pesos pré-treinados do sistema de *auto labeling*) sobre seu conjunto de validação.

Versão	Resolução	LR	Momento	Métricas			
				mAP@0.50	Precisão	Revocação	F1-score
AL1	480 × 480	10^{-5}	0,949	68,64%	0,52	**0,80**	0,63
AL2	480 × 480	10^{-7}	0,925	70,37%	**0,66**	0,65	0,65
AL3	704 × 704	10^{-6}	0,925	76,35%	0,57	0,78	0,66
AL4	704 × 704	10^{-5}	0,925	76,54%	0,64	0,76	0,69
AL5	704 × 704	10^{-4}	0,925	75,56%	0,60	**0,80**	0,69
AL6	704 × 704	10^{-4}	0,937	**76,96%**	0,64	0,78	**0,70**
AL7	704 × 704	10^{-5}	0,937	76,18%	0,64	0,78	**0,70**

entretanto, seus resultados sobre o conjunto de teste não se mantiveram no mesmo patamar, alcançando os 68,40% reportados. Tentou-se reduzir o valor de momento para 0,960 na versão 9 com o intuito de se alcançar um resultado semelhante ao da versão 8 para o conjunto de testes, entretanto, apesar de haver ganho nesse conjunto, o mesmo não foi o suficiente para justificar mais testes com o hiperparâmetro de momento.

4.2.2 Modelo utilizando os pesos pré-treinados do sistema de *auto labeling*

Como os resultados do modelo utilizando apenas os pesos pré-treinados sobre a base de dados MS COCO estavam abaixo do esperado, treinou-se um novo modelo utilizando os coeficientes treinados anteriormente para o sistema de *auto labeling*. Os resultados desse novo modelo estão sintetizados na Tabela 4.3. Apresenta-se o gráfico de treinamento da versão AL5 na Figura 4.2, pois essa obteve os melhores resultados. Os demais gráficos estão na Seção A.3 do Apêndice A. Conforme apresentado na Tabela Tabela 4.3, a versão AL1 alcançou um resultado inicial de mAP@0.50 próximo a do modelo anterior, entretanto, como comentado anteriormente, sofreu com instabilidade do gradiente. Dessa maneira, foram testados novos hiperparâmetros para tentar melhorar seus resultados.

A versão AL2 manteve a resolução da camada de entrada da rede e testou uma modificação nos hiperparâmetros de LR e momento, além de fazer uso da *flag* `-clear` da implementação da YOLOv4. Essa versão apresentou ganhos de mAP@0.50 e o melhor resultado na métrica de precisão. Todas as versões subsequentes fizeram uso da *flag* `-clear`. Em seguida, a versão AL3 testou o aumento da resolução das imagens na camada de entrada e da taxa de aprendizagem, obtendo ganhos significativos de mAP. As versões AL4 e AL5 testaram novos aumentos da LR, com o melhor resultado de mAP@0.50 sendo obtido com a taxa de aprendizagem igual a 10^{-5}. Não se testou o valor de LR igual a

Tabela 4.4: Resultados da validação cruzada *k-fold* da YOLOv4.

Fold	Métricas				
	mAP@0.50	Precisão	Revocação	F1-score	Tempo (h)
1	73,69%	0,46	0,93	0,61	6,16
2	74,66%	0,57	0,85	0,68	6,04
3	75,15%	0,59	0,84	0,70	5,98
4	82,00%	0,60	0,89	0,71	6,09
5	72,67%	0,43	0,89	0,58	6,02
Média	75,63%	0,53	0,88	0,66	6,06
Desvio padrão	3,29%	0,07	0,03	0,05	0,06

10^{-3} pois já se havia constatado anteriormente que esse valor levava à instabilidade no gradiente.

Por fim, testou-se o incremento no hiperparâmetro de momento nas versões AL6 e AL7. A versão AL6 obteve o melhor resultado de mAP@0.50, alcançando 76,96% nessa métrica, assim como o melhor F1-score, com 0,70. Esperava-se que a versão AL7 obteria os melhores resultados, uma vez que foi treinada com o que era até então o melhor valor para LR. Entretanto, isso não se concretizou, com seu resultado de mAP sendo marginalmente menor que o da sua predecessora. Dessa maneira, foram escolhidas as configurações da versão AL6 para o teste de validação cruzada *k-fold*.

4.2.3 Teste de validação cruzada *k-fold* da YOLOv4

O teste de validação cruzada *k-fold* para a YOLOv4 foi realizado conforme descrito na Subseção 3.5.2 utilizando os hiperparâmetros da versão AL6. Os resultados desse teste estão sintetizados na Tabela 4.4 e apresenta-se o gráfico de treinamento do *fold* 3 na Figura 4.3, por ser o conjunto sobre o qual as métricas mais se aproximam da média calculada. Os gráficos de treinamento dos demais *folds* estão apresentados na Seção A.4 do Apêndice A. É possível observar que os resultados de mAP@0.50 estão, em sua maioria, dentro do intervalo de dispersão calculado de $75,63\% \pm 3,29\%$, a exceção do *fold* 5 que, apesar de ser o melhor resultado obtido, é, claramente, um ponto fora da curva. Os resultados de precisão e F1-score obtidos foram abaixo do esperado, considerando o valor alcançado pela versão AL6. Por outro lado, os resultados de revocação foram melhores.

Dessa maneira, acredita-se que um modelo utilizando a arquitetura em questão seria viável para a detecção de macaúbas e babaçus em imagens obtidas por meio de UAV. Entretanto, a arquitetura da YOLOv4 em sua implementação no *framework* Darknet apresenta certas instabilidades, como é possível observar na Figura A.21 relativa ao treinamento do *fold* 2. Optou-se por se considerar esse *fold*, pois, apesar da instabilidade no gradiente a partir da época 2.000, os melhores resultados do sistema de *auto labeling* ocor-

riam após 1.000 épocas, aproximadamente, logo, o comportamento da rede está dentro do esperado. Outro problema da Darknet é a impossibilidade de se definir um parâmetro para tornar o treinamento determinístico, o que faz com que seja difícil de reproduzir os resultados de treinamento múltiplas vezes. Por outro lado, essa implementação da YOLOv4 já possui um *pipeline* integrado de *data augmentation*. Desse modo, conclui-se que é possível utilizar essa arquitetura para o trabalho em questão, entretanto, é necessário se atentar a seus aspectos negativos.

4.3 Modelo utilizando YOLOv9

Nos experimentos realizados com a arquitetura YOLOv9 os resultados mostraram-se promissores em todas a métricas a princípio. Os resultados desse modelo sobre seu conjunto de validação estão sintetizados na Tabela 4.5. A versão Y9-1 se sobressaiu nas métricas de mAP@0.50 e precisão, com resultados comparáveis aos melhores resultados de treinamento da YOLOv4. Apresentam-se os gráficos de treinamento, matriz de confusão e curva precisão-revocação (curva PR) dessa versão nas Figuras 4.4 a 4.6.

Tabela 4.5: Resultados de treinamento dos modelos utilizando a arquitetura YOLOv9 sobre seu conjunto de validação.

Versão	Otimizador	Épocas	Resolução	LR	Momento	Métricas			
						mAP@0.50	Precisão	Revocação	F1-score
Y9-1	AdamW	100	640 × 640	$7,14 \cdot 10^{-4}$	0,900	**74,9%**	**0,693**	0,726	0,709
Y9-2	AdamW	200	640 × 640	$7,14 \cdot 10^{-4}$	0,900	72,8%	0,639	0,730	0,681
Y9-3	SGD+Nesterov	100	640 × 640	$1,0 \cdot 10^{-2}$	0,937	72,0%	**0,693**	0,723	0,708
Y9-4	AdamW	200	640 × 640	$1,0 \cdot 10^{-4}$	0,937	69,9%	0,608	0,769	0,679
Y9-5	Adam	100	640 × 640	$1,0 \cdot 10^{-2}$	0,937	65,4%	0,587	0,730	0,651
Y9-6	AdamW	100	672 × 672	$7,14 \cdot 10^{-4}$	0,900	71,8%	0,642	0,781	0,705
Y9-7	AdamW	100	704 × 704	$7,14 \cdot 10^{-4}$	0,900	73,4%	0,628	**0,821**	**0,712**

As versões seguintes exploraram diferentes configurações de hiperparâmetros, visando alcançar melhores resultados. A versão Y9-2 constatou que o limiar de 100 épocas como máximo é o suficiente para se obter os melhores resultados. Em sequência, a Y9-3 testou um otimizador diferente (SGD associado ao gradiente acelerado de Nesterov) e, apesar de obter resultados semelhantes de precisão, revocação e F1-score aos do otimizador AdamW, seu resultado de mAP@0.50 foi inferior, demonstrando que o otimizador AdamW era melhor para o trabalho em questão. A Y9-4 averiguou uma estratégia de treinamento mais longo, com uma LR reduzida, o que se demonstrou ineficaz.

A versão Y9-5 testou o uso do otimizador Adam para o problema, entretanto, seu resultado foi significativamente inferior ao resultado do SGD associado ao gradiente de Nesterov, em condições semelhantes de LR e momento que, por sua vez, foi inferior aos resultados obtidos com o otimizador AdamW. Isso indica que esse último otimizador é o

Tabela 4.6: Resultados da validação cruzada *k-fold* da YOLOv9.

Fold	Métricas				
	mAP@0.50	Precisão	Revocação	F1-score	Tempo (h)
1	73,4%	0,658	0,791	0,718	0,769
2	74,3%	0,637	0,798	0,708	0,764
3	76,4%	0,592	0,873	0,706	0,770
4	72,0%	0,633	0,760	0,691	0,756
5	72,1%	0,749	0,662	0,703	0,778
6	72,8%	0,649	0,775	0,706	0,766
7	72,4%	0,695	0,656	0,675	0,775
8	66,9%	0,570	0,737	0,643	0,771
9	74,3%	0,654	0,792	0,716	0,771
10	71,9%	0,623	0,789	0,696	0,765
Média	72,7%	0,646	0,763	0,696	0,767
Desvio padrão	2,3%	0,048	0,062	0,022	0,006

melhor para o problema em questão. As versões Y9-6 e Y9-7 testaram resoluções maiores para a camada de entrada da rede. Foram obtidos os melhores resultados de revocação e F1-score dos modelos utilizando a YOLOv9. Por outro lado, esse resultado de F1-score é devido a um alto valor na métrica de revocação e acompanhado por um valor baixo na métrica de precisão. Dessa maneira, a versão Y9-1 é considerada a mais balanceada e superior dentre os modelos treinados com esta arquitetura.

4.3.1 Teste de validação cruzada *k-fold* da YOLOv9

O teste de validação cruzada *k-fold* para a YOLOv9 foi realizado conforme descrito na Subseção 3.6 utilizando os hiperparâmetros da versão Y9-1. Os resultados desse teste estão sintetizados na Tabela 4.6 e nas Figuras 4.7 a 4.9. Novamente, é possível observar que os valores obtidos para a métrica de mAP@0.50 estão dentro do intervalo de dispersão calculado de $72,7\% \pm 2,3\%$ em sua maioria, cujas exceções são os *folds* 3 e 8. Mais uma vez, o resultado médio da precisão foi abaixo do esperado, tomando por base o resultado obtido com a versão Y9-1. A média da revocação por outro lado, obteve um valor acima do esperado, ao passo que o F1-score médio ficou próximo do valor esperado para a rede. O destaque se dá pela métrica de tempo de treinamento, onde há uma redução de 87%, aproximadamente, em comparação com a YOLOv4.

Apresentam-se os gráficos de perda, precisão, revocação e mAP em função das épocas, a matriz de confusão e a curva PR relativos ao *fold* 6 nas Figuras 4.7 a 4.9, respectivamente. Esse *fold* foi escolhido por estar mais próximo da média das métricas apresentadas. Os gráficos, matrizes de confusão e curvas PR dos demais *folds* estão dispostos na Seção B.2 do Apêndice B. Os gráficos relativos ao *fold* 6 apresentam o comportamento esperado,

com a perda sendo reduzida ao longo das épocas ao passo que as métricas tendem a subir e estabilizar.

Um ponto importante a ser comentado foi observado na matriz de confusão das redes: elas tendem a marcar muitas instâncias do plano de fundo das imagens como babaçu. Assume-se que esse problema se dá em função das instâncias de babaçu que não foram marcadas nas imagens por estarem distorcidas. Como a imagem é reduzida de 2.000 × 2.000 px para 640 × 640 px na YOLOv9 (e para 704 × 704 na YOLOv4) perde-se os detalhes dessas distorções, como também é possível observar na Figura 3.3. Isso pode explicar a queda na performance da rede YOLOv4, que chegou a alcançar um mAP@0.50 de 86,82% no sistema da ferramenta de *auto labeling* e, subsequentemente, os resultados obtidos pela YOLOv9. Por outro lado, é possível observar que as redes dificilmente confundem instâncias de babaçu e macaúba entre si, o que indica que, mesmo após a redução das imagens, ainda há características suficientes para se distinguir essas espécies.

Os resultados de validação cruzada *k-fold* na métrica mAP@0.50 obtidos pelo modelo utilizando a YOLOv9 foram inferiores ao do modelo utilizando a YOLOv4 (com os pesos pré-treinados do sistema de *auto labeling*), 72,7% contra 76,96%, respectivamente. Entretanto, ainda há margem de crescimento para os resultados da YOLOv9, uma vez que essa arquitetura não possui um *pipeline* integrado de *data augmentation*, o que poderia gerar melhora em suas métricas. Outro ponto importante a ser levado em consideração é o fato que a YOLOv9 foi treinada apenas a partir de pesos pré-treinados sobre a base MS COCO, obtendo resultados melhores que o modelo de produção da YOLOv4 (que foi treinado de maneira similar). A implementação da YOLOv9 no *framework* Ultralytics também se prova vantajosa: são produzidas mais informações automaticamente (como a matriz de confusão do modelo e sua curva PR, entre outros), é possível configurar para que o treinamento seja determinístico (de modo que os resultados podem ser reproduzidos e devem se manter dentro da margem de erro calculada), o treinamento é mais rápido, o *framework* em questão é mais fácil de ser utilizado e sua documentação é mais atualizada. Dessa maneira, acredita-se que um modelo utilizando a YOLOv9 também seria adequado para a tarefa de detecção de macaúbas e babaçus em imagens obtidas por meio de UAV.

4.4 Discussão dos resultados

Em termos das métricas avaliadas, a YOLOv4 se sobressai nas métricas de mAP@0.50 (75,63% contra 72,7% da YOLOv9) e revocação (0,88 contra 0,763), ao passo que a YOLOv9 se destaca nos resultados de precisão (0,53 da YOLOv4 contra 0,646 da YOLOv9) e de F1-score (0,66 contra 0,696). Como a métrica de *mean average precision* leva em consideração a troca entre precisão e revocação, a YOLOv4 aparenta possuir uma vanta-

gem sobre a YOLOv9. Por outro lado, a YOLOv9 não teve acesso a nenhum *pipeline* de *data augmentation*, o que poderia melhorar seus resultados. Assim, para aplicações onde a precisão do modelo é primordial, a versão da YOLOv4 que fez uso dos pesos pré-treinados no sistema de auto labeling se sobressai no contexto deste trabalho.

Por outro lado, levando em consideração a parte prática de treinamento e até mesmo o contexto de produção de um modelo, a YOLOv9 se apresenta como mais vantajosa, uma vez que supera o modelo de produção da YOLOv4 em condições iguais (ao se treinar apenas a partir dos pesos pré-treinados no conjunto de dados MS COCO) com facilidade e é comparável ao modelo que utilizou os pesos pré-treinados do sistema de *auto labeling*. No contexto de produção de um sistema de ML, é preciso levar em conta que não necessariamente haveria o modelo de *auto labeling*. Por outro lado, ainda há a possibilidade de se gerar mais imagens de treinamento, por meio de técnicas de *data augmentation*, e melhorar os resultados da YOLOv9. Como comentado anteriormente, o *framework* da Ultralytics é mais fácil de ser utilizado e produz mais informações de maneira automatizada, como a matriz de confusão, que revelou detalhes importantes do comportamento das redes treinadas. Dessa maneira, a YOLOv9 também pode ser utilizada no contexto deste trabalho, uma vez que alcança resultados equiparáveis a de sua predecessora.

Comparando este trabalho com o de Jintasuttisak, Edirisinghe e Elbattay [19] que é o que mais se aproxima em termos de especificações, ainda há o que se melhorar, especialmente na métrica mAP@0.50, na qual os autores alcançaram melhor resultado de 92,34%. Porém, é preciso salientar que o contexto desta pesquisa se deu em áreas de ocorrência natural das espécies de palmeiras analisadas, ao passo que o trabalho supracitado se deu em áreas de plantio, que são mais simples. Além disso, este estudo lida com a detecção de duas espécies enquanto aquele lida apenas com uma. Assim, ao se analisar objetivamente o que foi realizado, ainda há melhorias a serem feitas, como no que diz respeito à anotação das imagens e a qualidade das marcações pelo modelo, mas este trabalho demonstra a possibilidade do uso de imagens obtidas a partir de UAV para detecção de palmeiras em ambientes naturais.

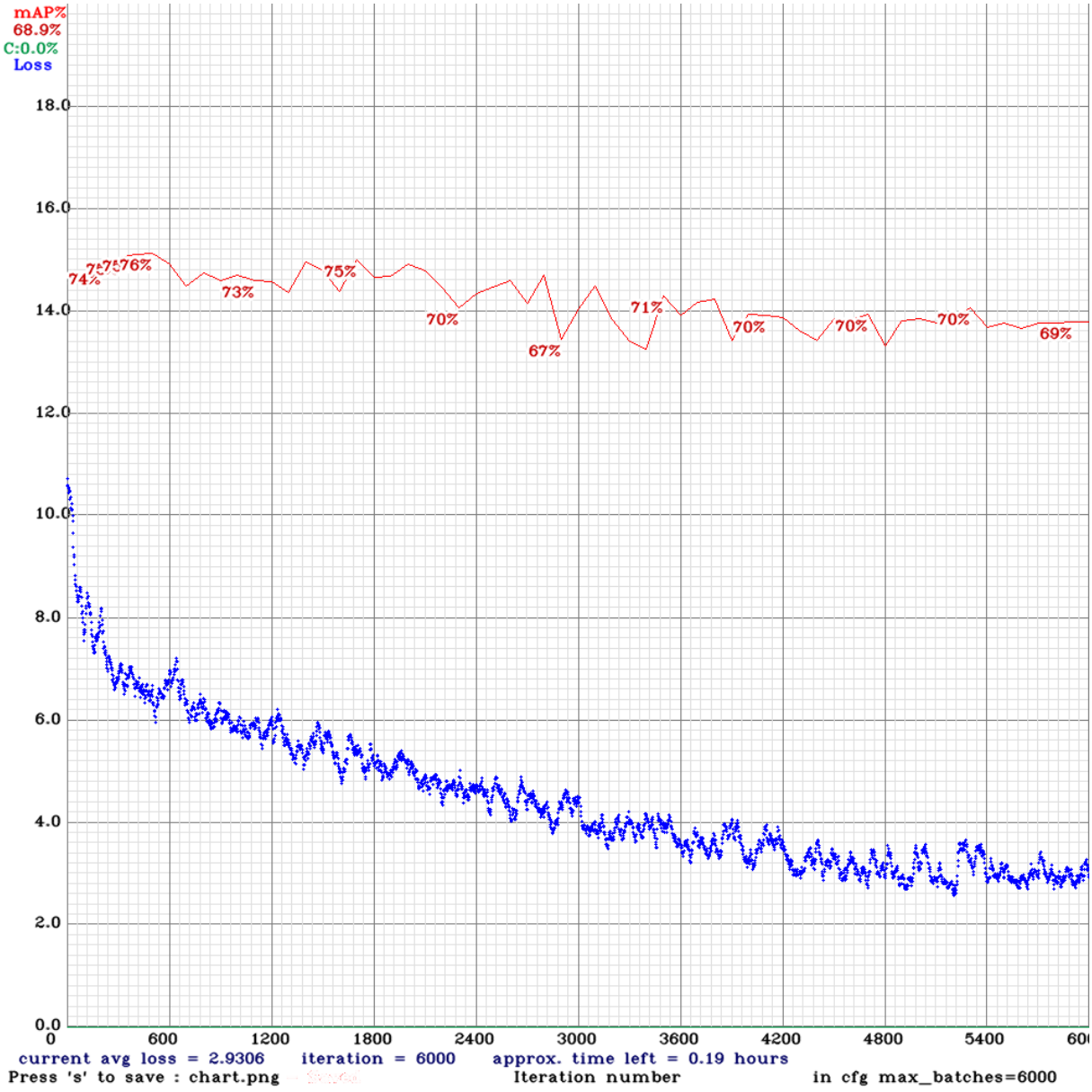

Figura 4.2: Gráfico da perda (em azul) e mAP@0.50 (em vermelho) em função das épocas relativo ao treinamento da versão AL5 do modelo de produção da YOLOv4 utilizando pesos pré-treinados do sistema de *auto labeling*.

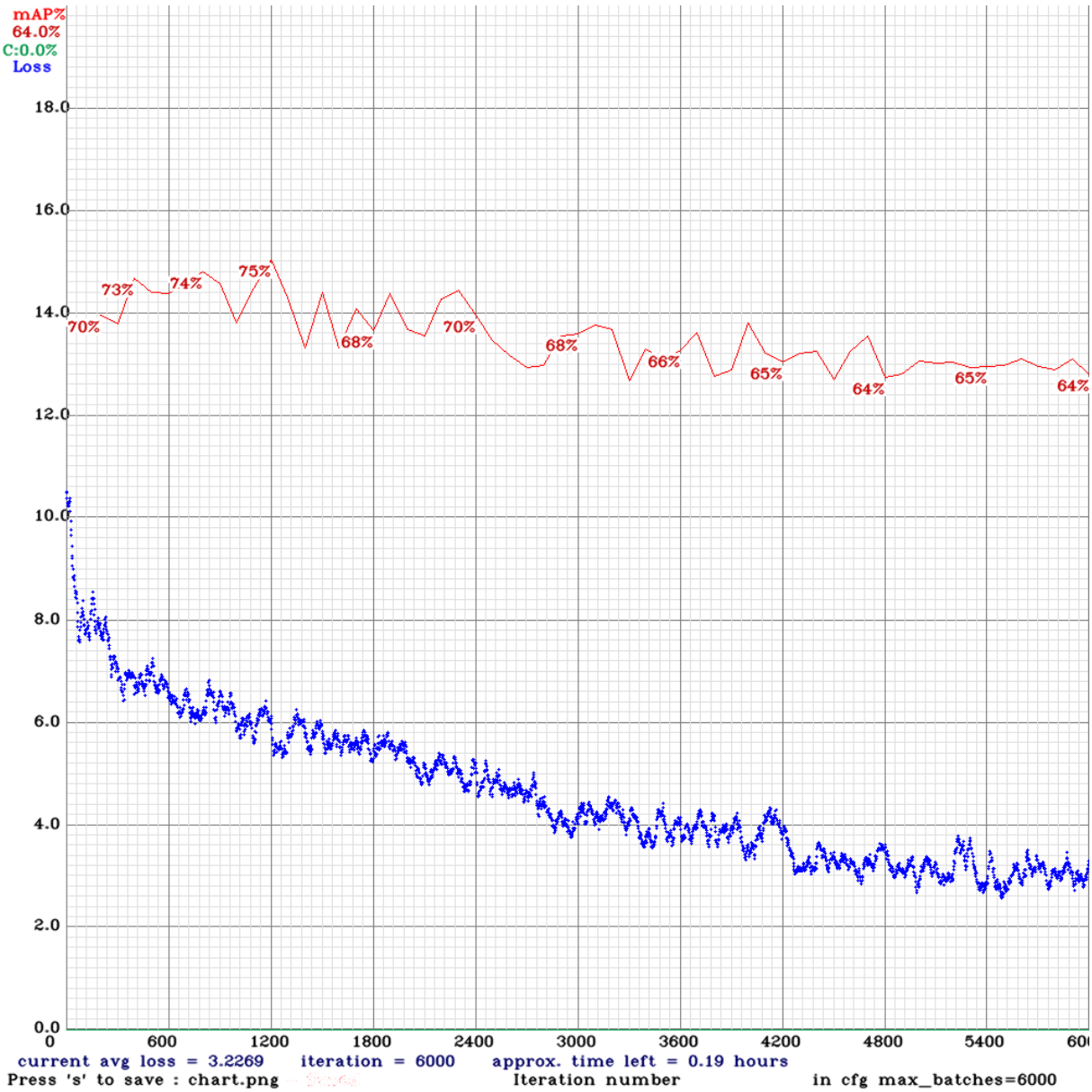

Figura 4.3: Gráfico da perda (em azul) e mAP@0.50 (em vermelho) em função das épocas relativo ao treinamento do *fold* 3 da validação cruzada *k-fold* da YOLOv4.

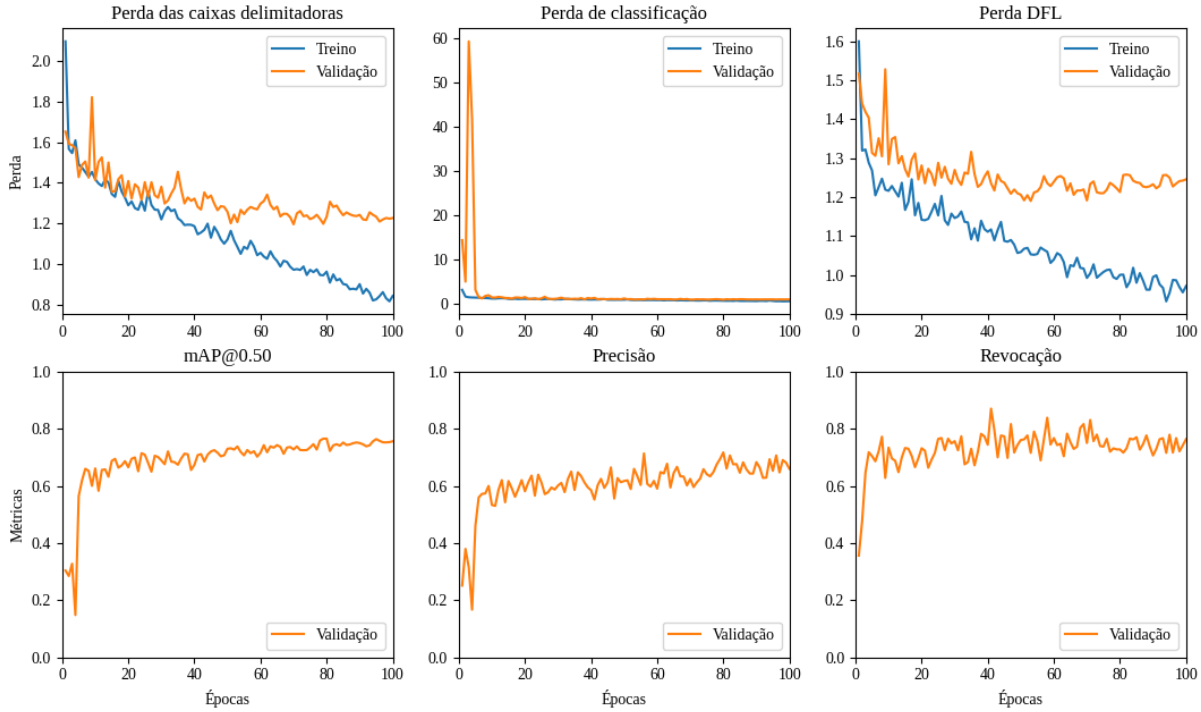

Figura 4.4: Gráficos das perdas das caixas delimitadoras, de classificação e DFL; mAP@ 0.50, precisão e revocação em função das épocas relativo ao treinamento da versão Y9-1 utilizando a YOLOv9.

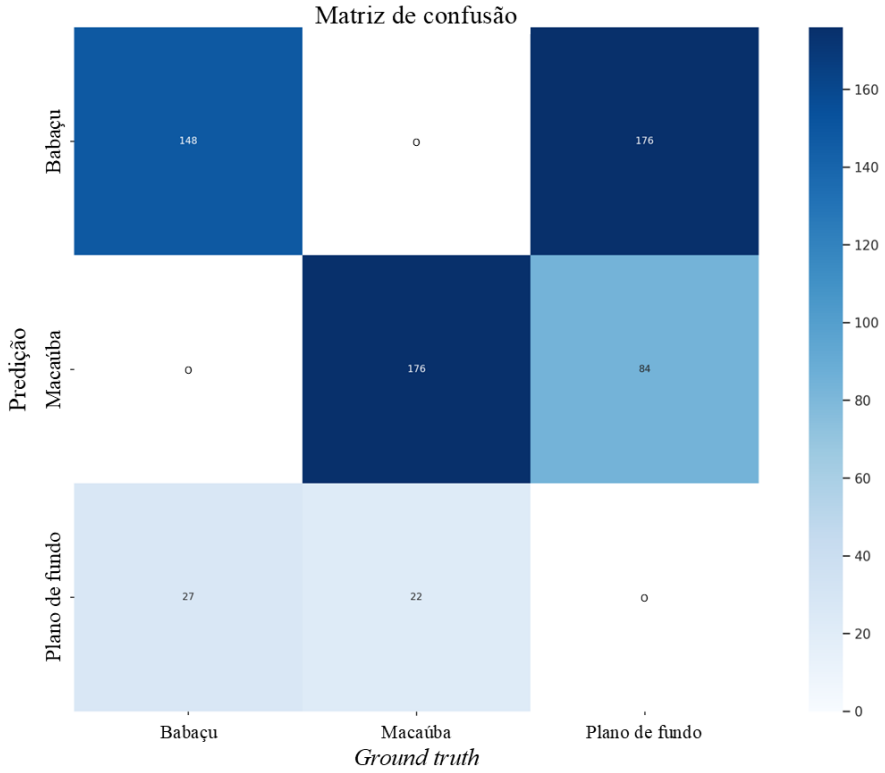

Figura 4.5: Matriz de confusão relativa à versão Y9-1 da YOLOv9.

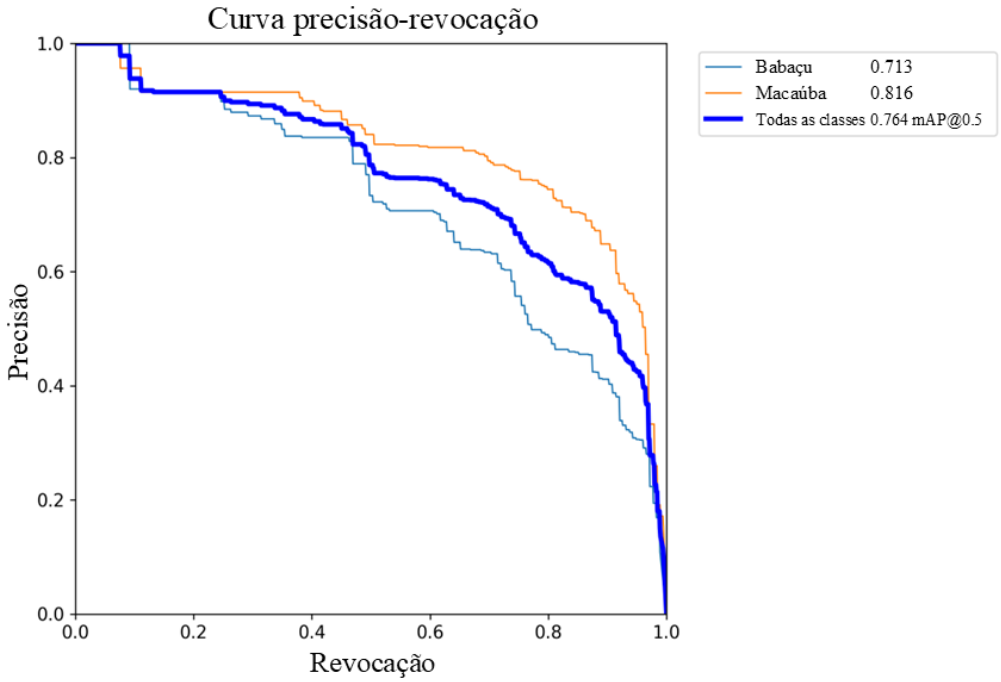

Figura 4.6: Curva PR relativa à versão Y9-1 da YOLOv9.

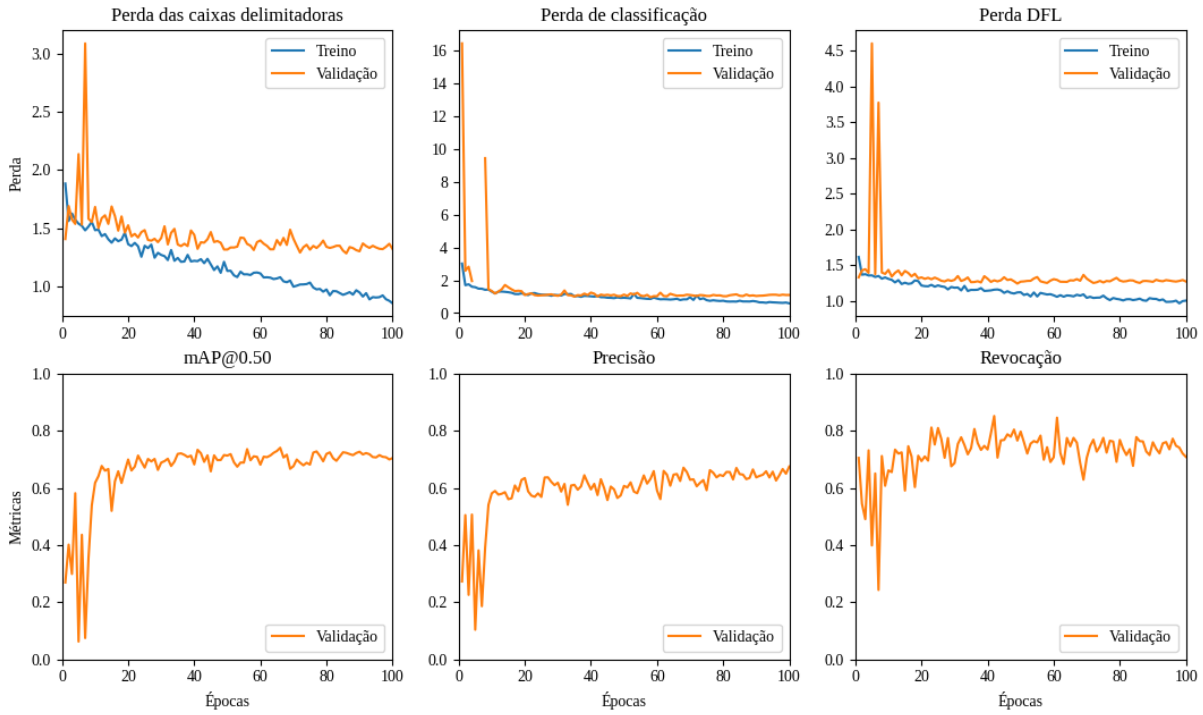

Figura 4.7: Gráficos das perdas das caixas delimitadoras, de classificação e DFL; mAP@ 0.50, precisão e revocação em função das épocas relativo ao treinamento do *fold* 6 da validação cruzada *k-fold* da YOLOv9.

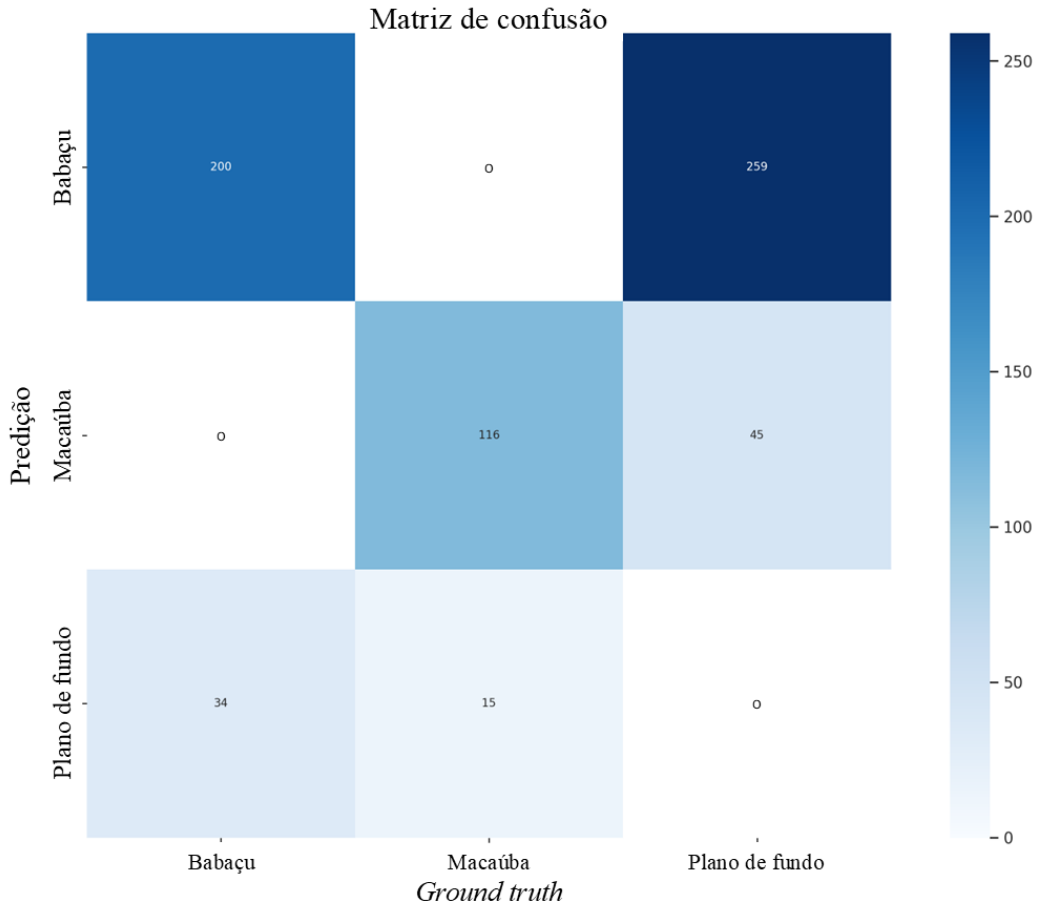

Figura 4.8: Matriz de confusão relativa ao treinamento do *fold* 6 da validação cruzada *k-fold* da YOLOv9.

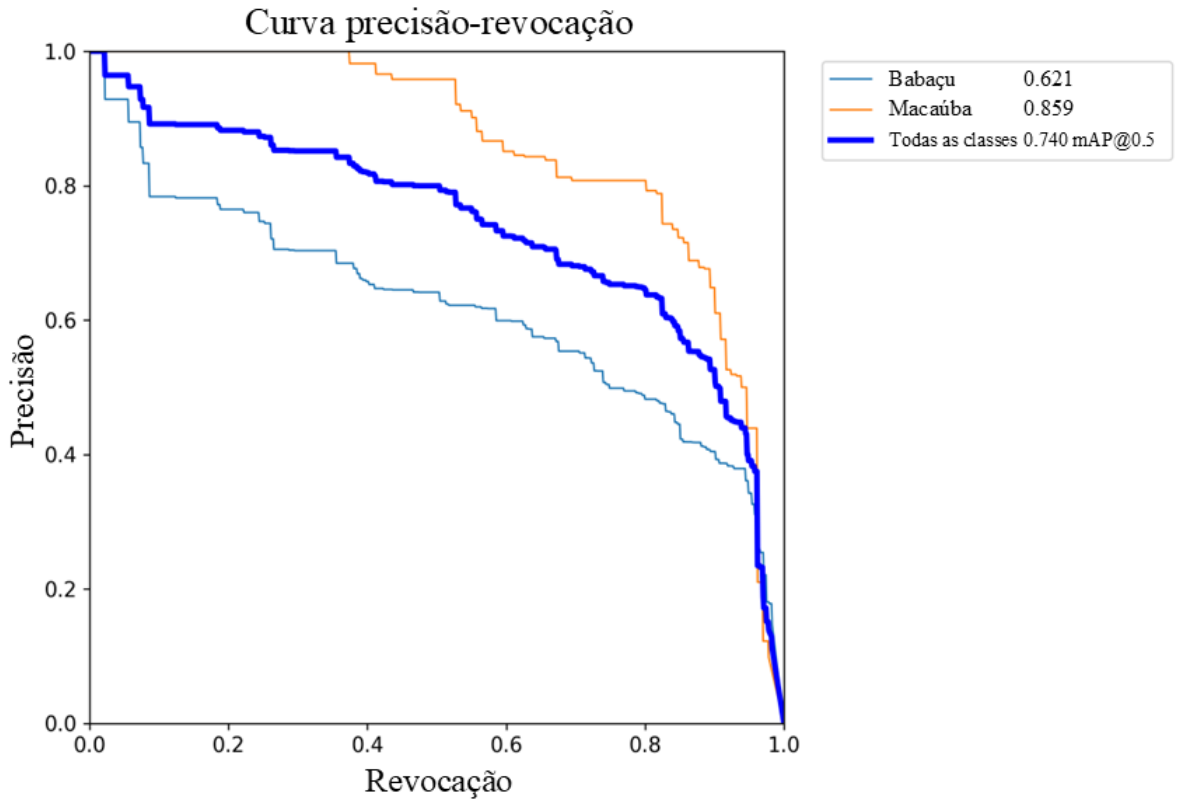

Figura 4.9: Curva PR relativa ao treinamento do *fold* 6 da validação cruzada *k-fold* da YOLOv9.

Capítulo 5

Conclusão

Neste trabalho foram apresentados dois modelos para detecção e diferenciação de espécimes de babaçu e macaúba em imagens obtidas a partir de UAV em ambientes naturais. Além disso, foram feitas as anotações do conjunto de dados de forma manual, auxiliada por um sistema de *auto labeling*. Esse último foi de grande ajuda para o estudo, dado a quantidade necessária de dados para treinar um modelo de DL. Apesar de suas marcações não serem perfeitas, o sistema fez grande parte do trabalho, necessitando de pequenos ajustes em suas anotações e deixando poucas instâncias sem marcações.

Os modelos treinados para detecção fizeram uso das arquiteturas YOLOv4 e YOLOv9, com a primeira rede obtendo os melhores resultados na métrica de mAP@0.50 (75,63% $\pm$ 3,29% e 72,7% $\pm$ 2,3%, respectivamente). É possível afirmar que a rede utilizando a arquitetura YOLOv9 é capaz de distinguir com facilidade as espécies estudadas. Apesar dos resultados promissores, ainda há melhorias a serem feitas, especialmente no que tange a qualidade das anotações das imagens. Se reconhece a falta de conhecimento a nível de especialista para tanto e são propostos dois caminhos a serem seguidos: retomar a estratégia de marcar todo tipo de palmeira que se assemelhasse a uma das espécies desejadas, independentemente do nível de distorção nos espécimes; ou reduzir a dimensão das imagens para tamanhos mais compatíveis com as camadas de entradas das arquiteturas utilizadas. O primeiro caminho se justifica devido a redução das imagens para as dimensões de 704 $\times$ 704 px ou 640 $\times$ 640 px, que elimina ou reduz o efeito das distorções nas imagens. O segundo visa fornecer mais dados aos modelos para que esses aprendam a reconhecer espécimes com menores níveis de distorção.

Por fim, recomenda-se o uso do modelo treinado a partir da arquitetura YOLOv4 utilizando pesos pré-treinados do sistema de *auto labeling* para aplicações onde a precisão das detecções é primordial. Para um modelo treinado apenas a partir das imagens marcadas, sugere-se o uso da arquitetura YOLOv9, uma vez que seus resultados superam aqueles da YOLOv4 quando ajustados a partir de pesos pré-treinados na base MS COCO. Além

disso, o treinamento da YOLOv9 é mais rápido e pode se beneficiar do uso de técnicas de *data augmentation* para melhorar seus resultados. O framework Ultralytics (responsável neste trabalho pela implementação da YOLOv9) também é de grande ajuda para a produção e análise dos modelos.

Referências

[1] Colombo, Carlos Augusto, Chorfi Berton, Luiz Henrique, Diaz, Brenda Gabriela e Ferrari, Roseli Aparecida: *Macauba: a promising tropical palm for the production of vegetable oil*. OCL, 25(1):D108, 2018. https://doi.org/10.1051/ocl/2017038. 1

[2] Cardoso, Alexandre Nunes, Gilmar Souza Santos, Simone Palma Favaro, Cristina Bobrowski Diniz e Humberto Umbelino de Sousa: *Extrativismo da macaúba na região do Cariri Cearense: comercialização e oportunidades. / Macauba extractivism at the Cariri Cearense region: commercialization and opportunities.* Brazilian Journal of Development, 6(5):25261–25279, May 2020. https://ojs.brazilianjournals.com.br/ojs/index.php/BRJD/article/view/9672. 1

[3] Vargas-Carpintero, Ricardo, Thomas Hilger, Johannes Mössinger, Roney Fraga Souza, Juan Carlos Barroso Armas, Karen Tiede e Iris Lewandowski: *Acrocomia spp.: neglected crop, ballyhooed multipurpose palm or fit for the bioeconomy? A review*. Agronomy for Sustainable Development, 41(6):75, Nov 2021, ISSN 1773-0155. https://doi.org/10.1007/s13593-021-00729-5. 1

[4] *O mundo mais sustentável com a macaúba*. **INOCAS**, 2019. Disponível em: https://www.inocas.com.br/. Acesso em: 24 de jul. de 2023. 1

[5] *A natureza é o início e o fim de tudo*. **S.Oleum**, 2022. Disponível em: https://soleum.com.br/. Acesso em 24 de jul. de 2023. 1

[6] Pires, Pâmela da Costa Lima *et al.*: *Análise de competitividade do sistema agroindustrial da macaúba (Acrocomia aculeata) nas regiões do norte de minas gerais e sul do Ceará, Brasil*. Tese de mestrado, Universidade Federal Fluminense, 2018. 1

[7] Crocomo, O. J. e M. Melo: *Acrocomia Species (Macauba Palm)*, páginas 3–17. Springer Berlin Heidelberg, Berlin, Heidelberg, 1996, ISBN 978-3-662-10617-4. https://doi.org/10.1007/978-3-662-10617-4_1. 1

[8] Ferrari, Roseli e Joaquim Azevedo Filho: *Macauba as promising substrate for crude oil and biodiesel production*. Journal of Agricultural Science and Technology, páginas 1119–1126, janeiro 2012. 1

[9] Morcote-Ríos, Gaspar e Rodrigo Bernal: *Remains of palms (palmae) at archaeological sites in the new world: A review*. The Botanical Review, 67(3):309–350, Jul 2001, ISSN 1874-9372. https://doi.org/10.1007/BF02858098. 2

[10] Mota, Clenilso Sehnen, Thais Roseli Corrêa, José Antonio Saraiva Grossi, Ariane Castricini e A da S Ribeiro: *Exploração sustentável da macaúba para produção de biodiesel: colheita, pós-colheita e qualidade dos frutos*. Informe Agropecuário, 32(265):41–50, 2011. 2

[11] Moreira, Sandro L.S., Cleverson V. Pires, Gustavo E. Marcatti, Ricardo H.S. Santos, Hewlley M.A. Imbuzeiro e Raphael B.A. Fernandes: *Intercropping of coffee with the palm tree, macauba, can mitigate climate change effects*. Agricultural and Forest Meteorology, 256-257:379–390, 2018, ISSN 0168-1923. https://www.sciencedirect.com/science/article/pii/S0168192318301096. 2

[12] Al-helaly, Emad Ali e Noor Ali Al-Helaly: *A count of palm trees from satellite image*, 2018. 2, 6

[13] Zortea, Maciel, Marcelo Nery, Bernardo Ruga, Lara B. Carvalho e Adriano C. Bastos: *Oil-palm tree detection in aerial images combining deep learning classifiers*. Em *IGARSS 2018 - 2018 IEEE International Geoscience and Remote Sensing Symposium*, páginas 657–660, 2018. 2, 8

[14] Zheng, Juepeng, Weijia Li, Maocai Xia, Runmin Dong, Haohuan Fu e Shuai Yuan: *Large-scale oil palm tree detection from high-resolution remote sensing images using Faster-RCNN*. Em *IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium*, páginas 1422–1425, 2019. 2, 6

[15] Mubin, Nurulain Abd, Eiswary Nadarajoo, Helmi Zulhaidi Mohd Shafri e Alireza Hamedianfar: *Young and mature oil palm tree detection and counting using convolutional neural network deep learning method*. International Journal of Remote Sensing, 40(19):7500–7515, 2019. https://doi.org/10.1080/01431161.2019.1569282. 2, 6, 7

[16] Culman, María, Andrés C. Rodríguez, Jan Dirk Wegner, Stephanie Delalieux e Ben Somers: *Deep learning for sub-pixel palm tree classification using spaceborne Sentinel-2 imagery*. Em Neale, Christopher M. U. e Antonino Maltese (editores): *Remote Sensing for Agriculture, Ecosystems, and Hydrology XXIII*, volume 11856, página 118560E. International Society for Optics and Photonics, SPIE, 2021. https://doi.org/10.1117/12.2599861. 2, 6, 7

[17] Ferreira, Matheus Pinheiro, Danilo Roberti Alves de Almeida, Daniel de Almeida Papa, Juliano Baldez Silva Minervino, Hudson Franklin Pessoa Veras, Arthur Formighieri, Caio Alexandre Nascimento Santos, Marcio Aurélio Dantas Ferreira, Evandro Orfanó Figueiredo e Evandro José Linhares Ferreira: *Individual tree detection and species classification of amazonian palms using uav images and deep learning*. Forest Ecology and Management, 475:118397, 2020, ISSN 0378-1127. https://www.sciencedirect.com/science/article/pii/S037811272031166X. 2, 8

[18] Tagle Casapia, Ximena, Lourdes Falen, Harm Bartholomeus, Rodolfo Cárdenas, Gerardo Flores, Martin Herold, Eurídice N. Honorio Coronado e Timothy R. Baker:

Identifying and quantifying the abundance of economically important palms in tropical moist forest using uav imagery. Remote Sensing, 12(1), 2020, ISSN 2072-4292. https://www.mdpi.com/2072-4292/12/1/9. 2, 9, 20

[19] Jintasuttisak, Thani, Eran Edirisinghe e Ali Elbattay: *Deep neural network based date palm tree detection in drone imagery*. Computers and Electronics in Agriculture, 192:106560, 2022, ISSN 0168-1699. https://www.sciencedirect.com/science/article/pii/S0168169921005779. 2, 9, 39

[20] Yarak, Kanitta, Apichon Witayangkurn, Kunnaree Kritiyutanont, Chomchanok Arunplod e Ryosuke Shibasaki: *Oil palm tree detection and health classification on high-resolution imagery using deep learning*. Agriculture, 11(2), 2021, ISSN 2077-0472. https://www.mdpi.com/2077-0472/11/2/183. 2, 9

[21] Zaidi, Syed Sahil Abbas, Mohammad Samar Ansari, Asra Aslam, Nadia Kanwal, Mamoona Asghar e Brian Lee: *A survey of modern deep learning based object detection models*. Digital Signal Processing, 126:103514, 2022, ISSN 1051-2004. https://www.sciencedirect.com/science/article/pii/S1051200422001312. 3

[22] Goodfellow, Ian, Yoshua Bengio e Aaron Courville: *Deep Learning*, capítulo 1, páginas 1–2. MIT Press, Cambridge, MA, USA, 2016. http://www.deeplearningbook.org. 3, 10, 12

[23] Aggarwal, Charu C.: *Neural Networks and Deep Learning*. Springer, Cham, first edição, 2018, ISBN 978-3-319-94462-3978-3-319-94462-3. 3

[24] Bochkovskiy, Alexey, Chien Yao Wang e Hong Yuan Mark Liao: *YOLOv4: Optimal speed and accuracy of object detection*, 2020. 3, 9, 13

[25] Wang, Chien Yao, I Hau Yeh e Hong Yuan Mark Liao: *Yolov9: Learning what you want to learn using programmable gradient information*, 2024. 3, 14

[26] Xia, Gui Song, Jingwen Hu, Fan Hu, Baoguang Shi, Xiang Bai, Yanfei Zhong, Liangpei Zhang e Xiaoqiang Lu: *Aid: A benchmark data set for performance evaluation of aerial scene classification*. IEEE Transactions on Geoscience and Remote Sensing, 55(7):3965–3981, 2017. 5

[27] Xia, Gui Song, Xiang Bai, Jian Ding, Zhen Zhu, Serge Belongie, Jiebo Luo, Mihai Datcu, Marcello Pelillo e Liangpei Zhang: *Dota: A large-scale dataset for object detection in aerial images*. Em *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 5

[28] Cao, Yaru, Zhijian He, Lujia Wang, Wenguan Wang, Yixuan Yuan, Dingwen Zhang, Jinglin Zhang, Pengfei Zhu, Luc Van Gool, Junwei Han, Steven Hoi, Qinghua Hu, Ming Liu, Chong Cheng, Fanfan Liu, Guojin Cao, Guozhen Li, Hongkai Wang, Jianye He, Junfeng Wan, Qi Wan, Qi Zhao, Shuchang Lyu, Wenzhe Zhao, Xiaoqiang Lu, Xingkui Zhu, Yingjie Liu, Yixuan Lv, Yujing Ma, Yuting Yang, Zhe Wang, Zhenyu Xu, Zhipeng Luo, Zhimin Zhang, Zhiguang Zhang, Zihao Li e Zixiao Zhang: *Visdrone-det2021: The vision meets drone object detection challenge results*. Em *2021*

IEEE/CVF International Conference on Computer Vision Workshops (ICCVW), páginas 2847–2854, 2021. 5

[29] Lin, Chih Wei, Qilu Ding, Wei Hao Tu, Jia Hang Huang e Jin Fu Liu: *Fourier dense network to conduct plant classification using uav-based optical images*. IEEE Access, 7:17736–17749, 2019. 6

[30] Morales, Giorgio, Guillermo Kemper, Grace Sevillano, Daniel Arteaga, Ivan Ortega e Joel Telles: *Automatic segmentation of mauritia flexuosa in unmanned aerial vehicle (uav) imagery using deep learning*. Forests, 9(12), 2018, ISSN 1999-4907. `https://www.mdpi.com/1999-4907/9/12/736`. 6

[31] Bouguettaya, Abdelmalek, Hafed Zarzour, Ahmed Kechida e Amine Mohammed Taberkit: *Deep learning techniques to classify agricultural crops through uav imagery: A review*. Neural Computing and Applications, 34(12):9511–9536, 2022. 6

[32] Zeiler, Matthew D. e Rob Fergus: *Visualizing and understanding convolutional networks*. Em Fleet, David, Tomas Pajdla, Bernt Schiele e Tinne Tuytelaars (editores): *Computer Vision – ECCV 2014*, páginas 818–833, Cham, 2014. Springer International Publishing, ISBN 978-3-319-10590-1. 6

[33] Chatfield, Ken, Karen Simonyan, Andrea Vedaldi e Andrew Zisserman: *Return of the devil in the details: Delving deep into convolutional nets*. CoRR, abs/1405.3531, 2014. `http://arxiv.org/abs/1405.3531`. 6

[34] Simonyan, Karen e Andrew Zisserman: *Very deep convolutional networks for large-scale image recognition*, 2015. 6, 9

[35] LeCun, Y., L. Bottou, Y. Bengio e P. Haffner: *Gradient-based learning applied to document recognition*. Proceedings of the IEEE, 86(11):2278–2324, 1998. 7

[36] He, Kaiming, Xiangyu Zhang, Shaoqing Ren e Jian Sun: *Deep residual learning for image recognition*. Em *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, páginas 770–778, 2016. 7, 9

[37] Redmon, Joseph e Ali Farhadi: *YOLOv3: An incremental improvement*, 2018. 9, 13

[38] Liu, Wei, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng Yang Fu e Alexander C. Berg: *Ssd: Single shot multibox detector*. Em Leibe, Bastian, Jiri Matas, Nicu Sebe e Max Welling (editores): *Computer Vision – ECCV 2016*, páginas 21–37, Cham, 2016. Springer International Publishing, ISBN 978-3-319-46448-0. 9

[39] Munawar, Muhammad Rizwan, Glenn Jocher, Lakshantha Dissanayake, Burhan Qaddoumi e Sergiu Waxmann: *YOLOv5: A state-of-the-art real-time object detection system*. **Ultralytics YOLO Docs**, 2023. Disponível em: `https://docs.ultralytics.com/yolov5/`. Acesso em: 5 jun. de 2024. 9

[40] Ren, Shaoqing, Kaiming He, Ross Girshick e Jian Sun: *Faster R-CNN: Towards real-time object detection with region proposal networks*. Em Cortes, C., N. Lawrence, D. Lee, M. Sugiyama e R. Garnett (editores): *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc., 2015. `https://proceedings.neurips.cc/paper_files/paper/2015/file/14bfa6bb14875e45bba028a21ed38046-Paper.pdf`. 9, 13

[41] Géron, Aurélien: *Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*. O'Reilly Media, Incorporated, second edição, 2019, ISBN 9781492032649. `https://www.oreilly.com/library/view/hands-on-machine-learning/9781492032632/`. 10, 15

[42] Cauchy, Augustin Louis: *ANALYSE MATHÉMATIQUE. – Méthodc générale pour la résolution des systèmes d'équations simultanées*, página 399–402. Cambridge Library Collection - Mathematics. Cambridge University Press, 2009. 11

[43] Polyak, Boris: *Some methods of speeding up the convergence of iteration methods*. Ussr Computational Mathematics and Mathematical Physics, 4:1–17, dezembro 1964. 11

[44] Nesterov, Yurii: *A method for solving the convex programming problem with convergence rate* $O(1/k^2)$. Proceedings of the USSR Academy of Sciences, 269:543–547, 1983. 11

[45] Kingma, Diederik P. e Jimmy Ba: *Adam: A method for stochastic optimization*, 2017. `https://arxiv.org/abs/1412.6980`. 12

[46] Tieleman, Tijmen e Geoffrey Hinton: *Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude*. COURSERA: Neural networks for machine learning, 4(2):26–31, 2012. 12

[47] Wilson, Ashia C., Rebecca Roelofs, Mitchell Stern, Nathan Srebro e Benjamin Recht: *The marginal value of adaptive gradient methods in machine learning*, 2018. 12, 26

[48] Zou, Zhengxia, Keyan Chen, Zhenwei Shi, Yuhong Guo e Jieping Ye: *Object detection in 20 years: A survey*. Proceedings of the IEEE, 111(3):257–276, 2023. 12

[49] Girshick, Ross, Jeff Donahue, Trevor Darrell e Jitendra Malik: *Rich feature hierarchies for accurate object detection and semantic segmentation*. Em *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014. 13

[50] Girshick, Ross: *Fast R-CNN*. Em *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, December 2015. 13

[51] Redmon, Joseph, Santosh Divvala, Ross Girshick e Ali Farhadi: *You only look once: Unified, real-time object detection*. Em *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 13

[52] Hoiem, Derek, Santosh K Divvala e James H Hays: *Pascal voc 2008 challenge*. World Literature Today, 24(1), 2009. 13

[53] Redmon, Joseph e Ali Farhadi: *YOLO9000: Better, faster, stronger*. Em *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 13

[54] Lin, Tsung Yi, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár e C. Lawrence Zitnick: *Microsoft coco: Common objects in context*. Em Fleet, David, Tomas Pajdla, Bernt Schiele e Tinne Tuytelaars (editores): *Computer Vision – ECCV 2014*, páginas 740–755, Cham, 2014. Springer International Publishing, ISBN 978-3-319-10602-1. 13

[55] Wang, Chien Yao, Alexey Bochkovskiy e Hong Yuan Mark Liao: *Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors*. Em *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, páginas 7464–7475, June 2023. 14

[56] Lin, Zhihao, Yongtao Wang, Jinhe Zhang e Xiaojie Chu: *Dynamicdet: A unified dynamic architecture for object detection*. Em *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, páginas 6282–6291, June 2023. 14

[57] Yang, Suorong, Weikang Xiao, Mengchen Zhang, Suhan Guo, Jian Zhao e Furao Shen: *Image data augmentation for deep learning: A survey*, 2023. 15

[58] Shorten, Connor e Taghi M. Khoshgoftaar: *A survey on image data augmentation for deep learning*. Journal of Big Data, 6(1):60, Jul 2019, ISSN 2196-1115. `https://doi.org/10.1186/s40537-019-0197-0`. 15

[59] Zhuang, Fuzhen, Zhiyuan Qi, Keyu Duan, Dongbo Xi, Yongchun Zhu, Hengshu Zhu, Hui Xiong e Qing He: *A comprehensive survey on transfer learning*. Proceedings of the IEEE, 109(1):43–76, 2021. 15

[60] Deng, Jia, Wei Dong, Richard Socher, Li Jia Li, Kai Li e Li Fei-Fei: *Imagenet: A large-scale hierarchical image database*. Em *2009 IEEE Conference on Computer Vision and Pattern Recognition*, páginas 248–255, 2009. 15

[61] Fredriksson, Teodor, Jan Bosch e Helena Olsson: *Machine learning models for automatic labeling: A systematic literature review*. Em *Proceedings of the 15th International Conference on Software Technologies - Volume 1: ICSOFT,*, páginas 552–561. INSTICC, SciTePress, 2020, ISBN 978-989-758-443-5. 15

[62] Alvares, Clayton Alcarde, José Leonardo Stape, José Luiz and Sentelhas, Paulo Cesar and de Moraes Gonçalves e Gerd Sparovek: *Köppen's climate classification map for brazil*. Meteorologische Zeitschrift, 22(6):711–728, dezembro 2013. `http://dx.doi.org/10.1127/0941?2948/2013/0507`. 16

[63] IBGE: *Manual técnico de pedologia*. Rio de Janeiro, 2ª edição, 2007. 16

[64] Tadono, T., H. Nagai, H. Ishida, F. Oda, S. Naito, K. Minakawa e H. Iwamoto: *Generation of the 30 M-Mesh Global Digital Surface Model by Alos Prism*. ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 41B4:157–162, junho 2016. 16

[65] Charette, Stéphane: *Programming Comments - Darknet FAQ*. **Darknet/YOLO FAQ**, 2024. Disponível em: `https://www.ccoderun.ca/programming/darknet_faq/#time_to_train`. Acesso em: 9 de ago. de 2024. 18

[66] *Label Studio Documentation — Object detection in images with Label Studio and MMDetection*. **Label Studio Docs.** Disponível em: `https://labelstud.io/tutorials/mmdetection-3`. Acesso em: 9 de set. de 2024. 19

Apêndice A

Gráficos de treinamento da YOLOv4

A.1 Gráficos relativos ao treinamento da ferramenta de *auto labeling*

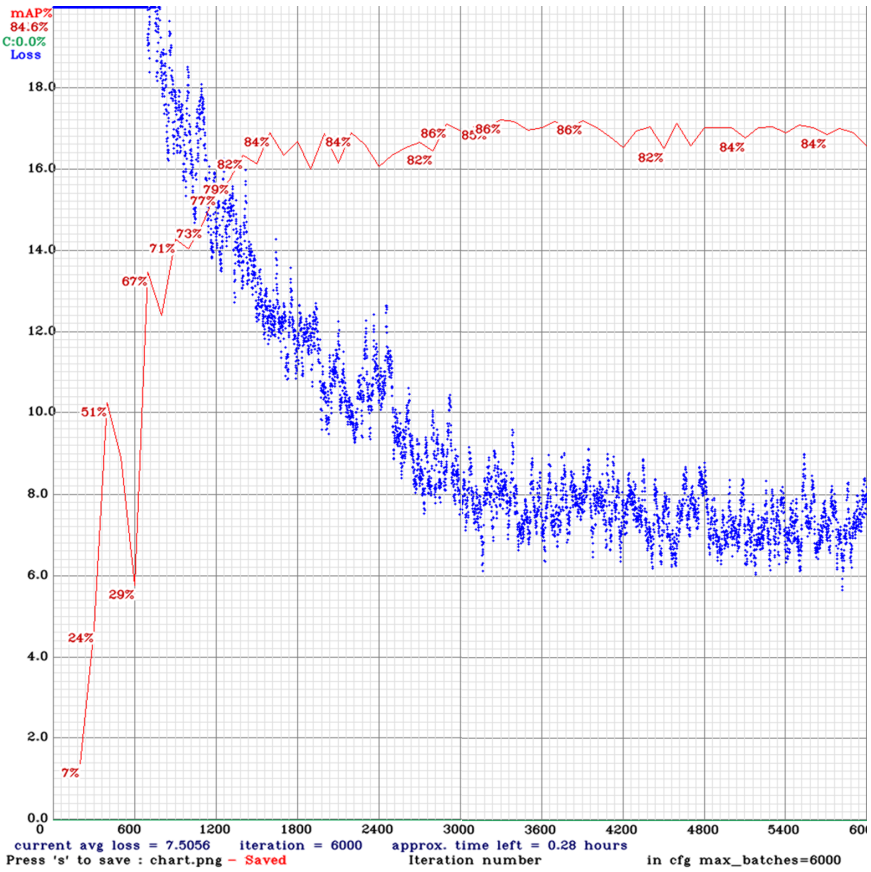

Figura A.1: Gráfico da perda (em azul) e mAP@0.50 (em vermelho) em função das épocas relativo ao treinamento da versão 1 do modelo de *auto labeling* utilizando a arquitetura YOLOv4.

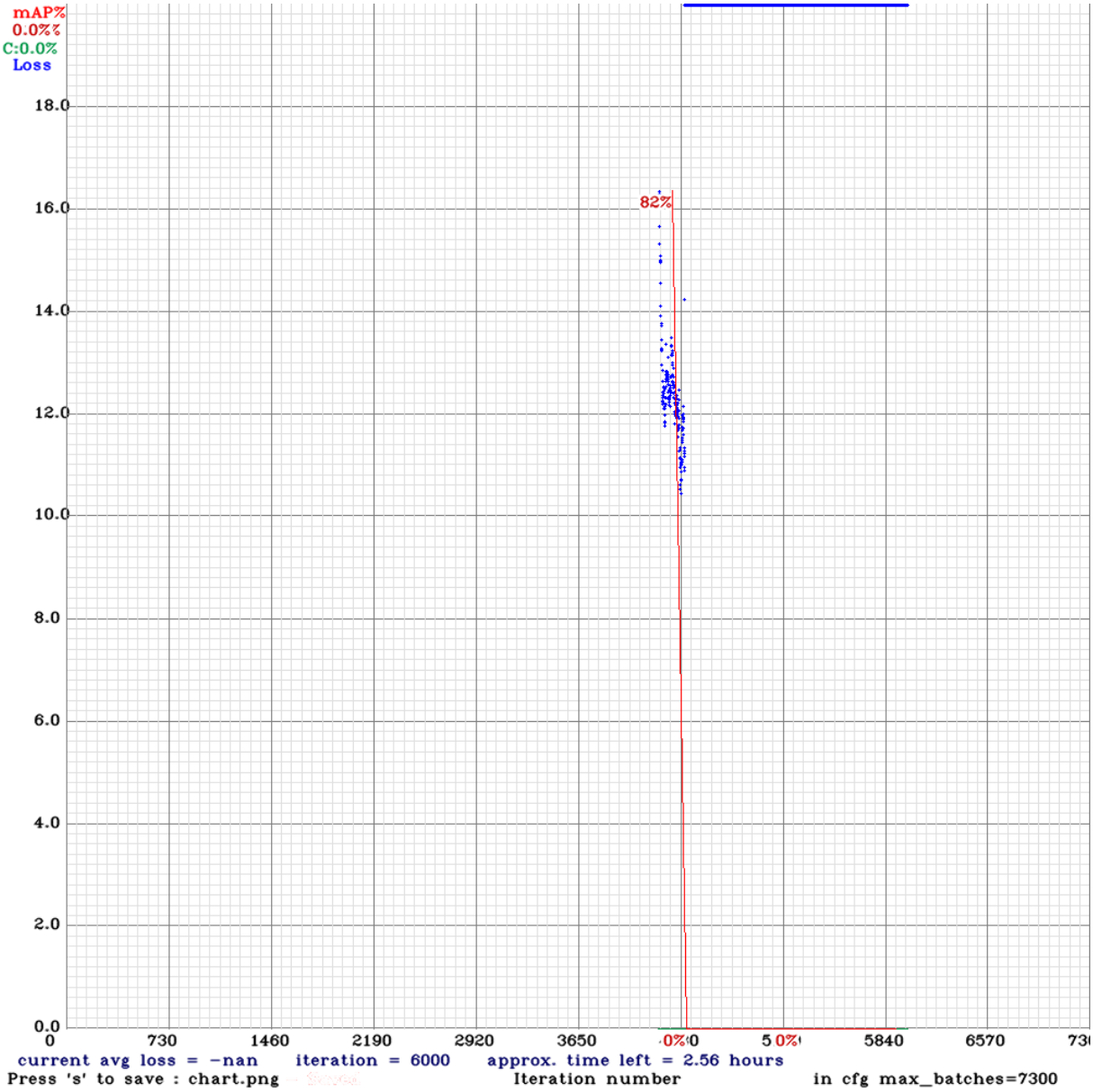

Figura A.2: Gráfico da perda (em azul) e mAP@0.50 (em vermelho) em função das épocas relativo ao treinamento da versão 2 do modelo de *auto labeling* utilizando a arquitetura YOLOv4.

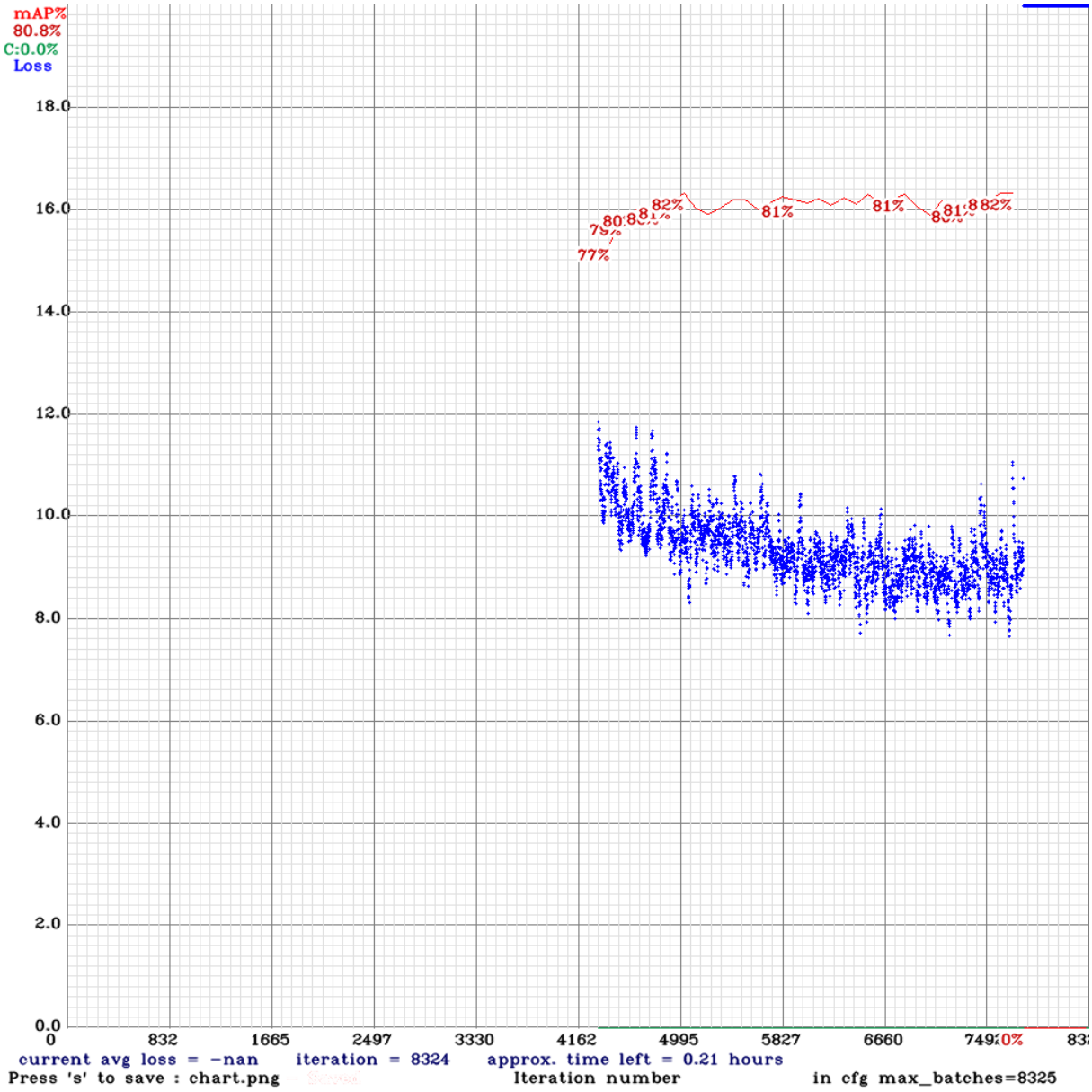

Figura A.3: Gráfico da perda (em azul) e mAP@0.50 (em vermelho) em função das épocas relativo ao treinamento da versão 3 do modelo de *auto labeling* utilizando a arquitetura YOLOv4.

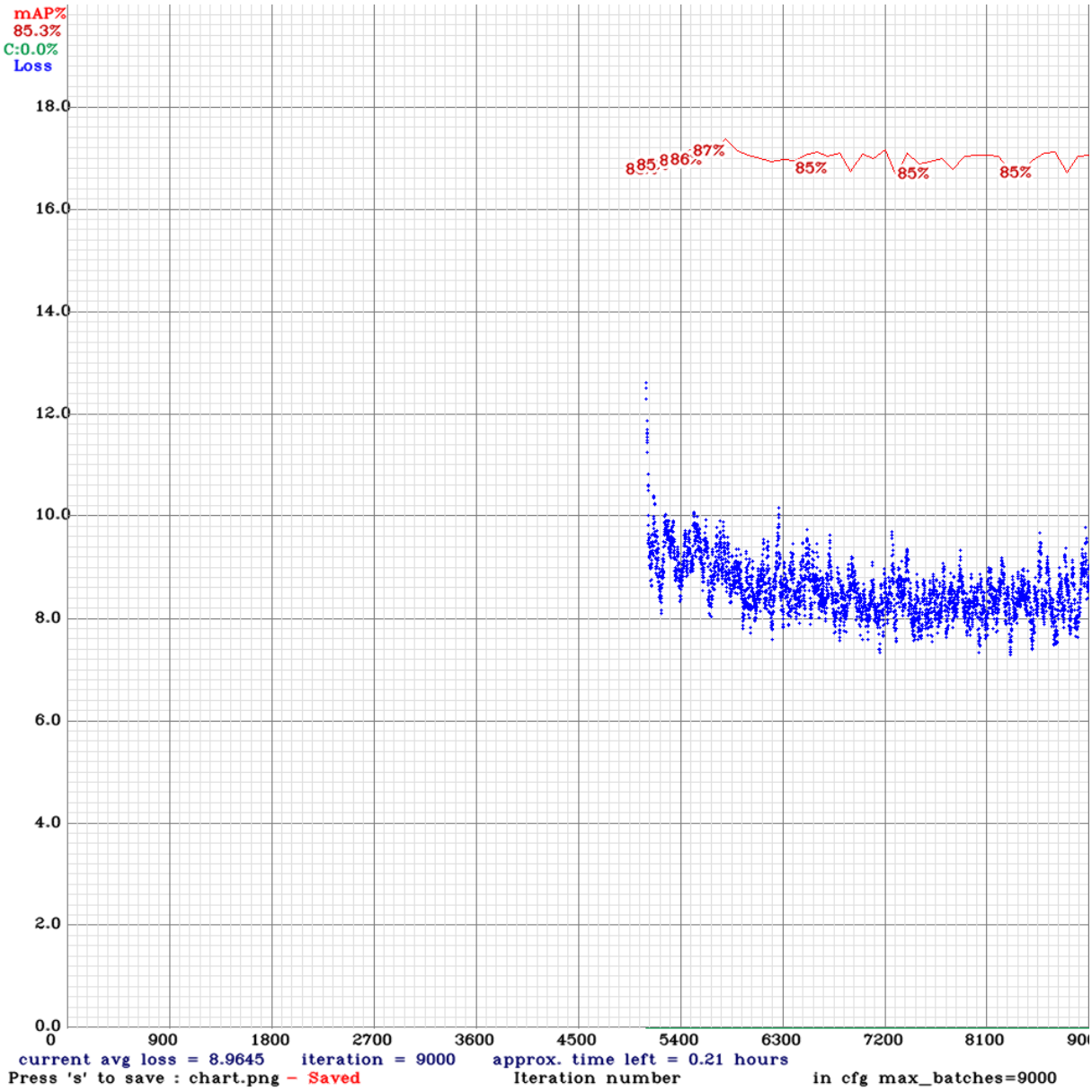

Figura A.4: Gráfico da perda (em azul) e mAP@0.50 (em vermelho) em função das épocas relativo ao treinamento da versão 4 do modelo de *auto labeling* utilizando a arquitetura YOLOv4.

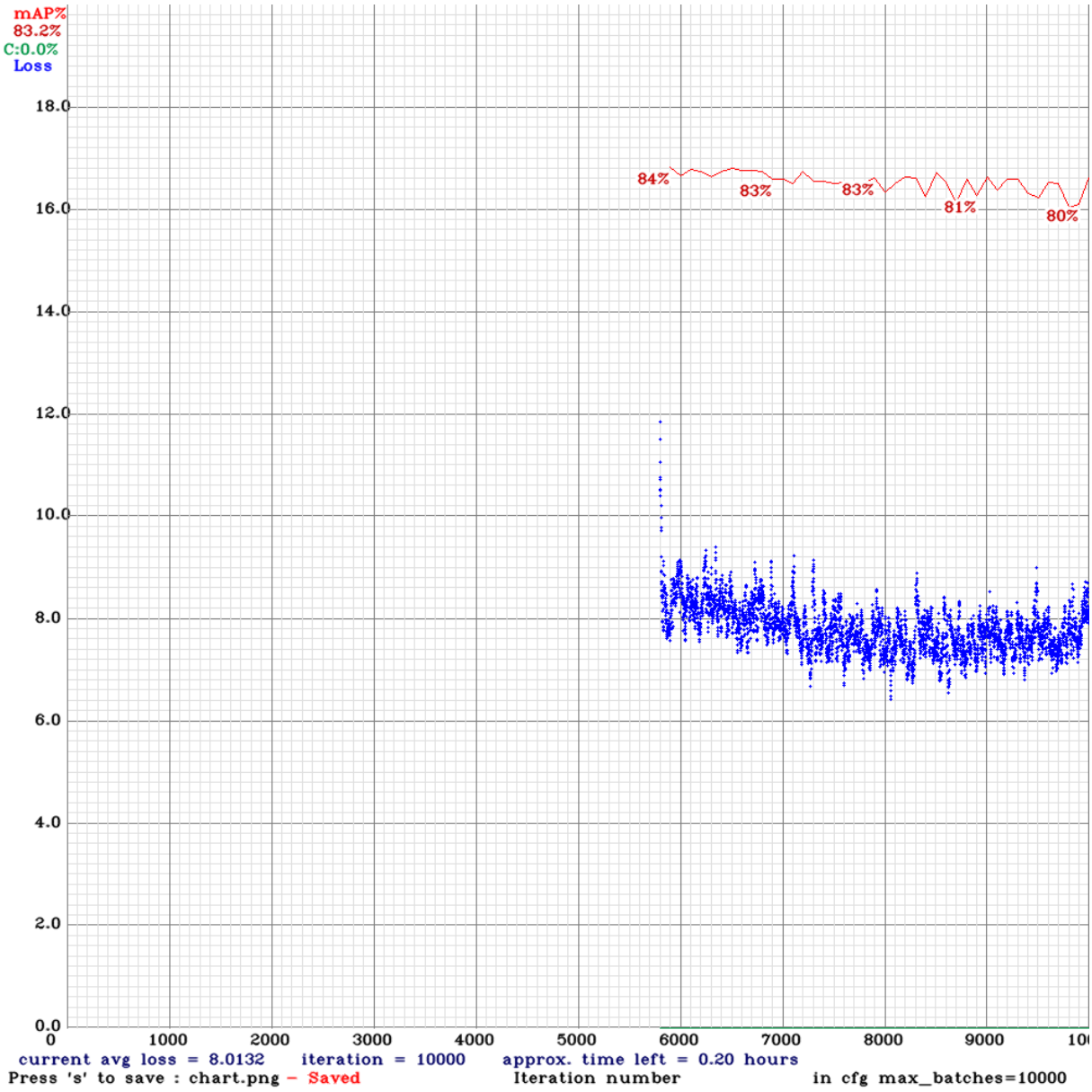

Figura A.5: Gráfico da perda (em azul) e mAP@0.50 (em vermelho) em função das épocas relativo ao treinamento da versão 5 do modelo de *auto labeling* utilizando a arquitetura YOLOv4.

A.2 Gráficos relativos ao treinamento do modelo de produção utilizando pesos pré-treinados sobre a base de dados MS COCO

O gráfico relativo a versão 3 do modelo de produção da YOLOv4 utilizando pesos pré-treinados sobre a base de dados MS COCO não foi produzido devido a instabilidade do

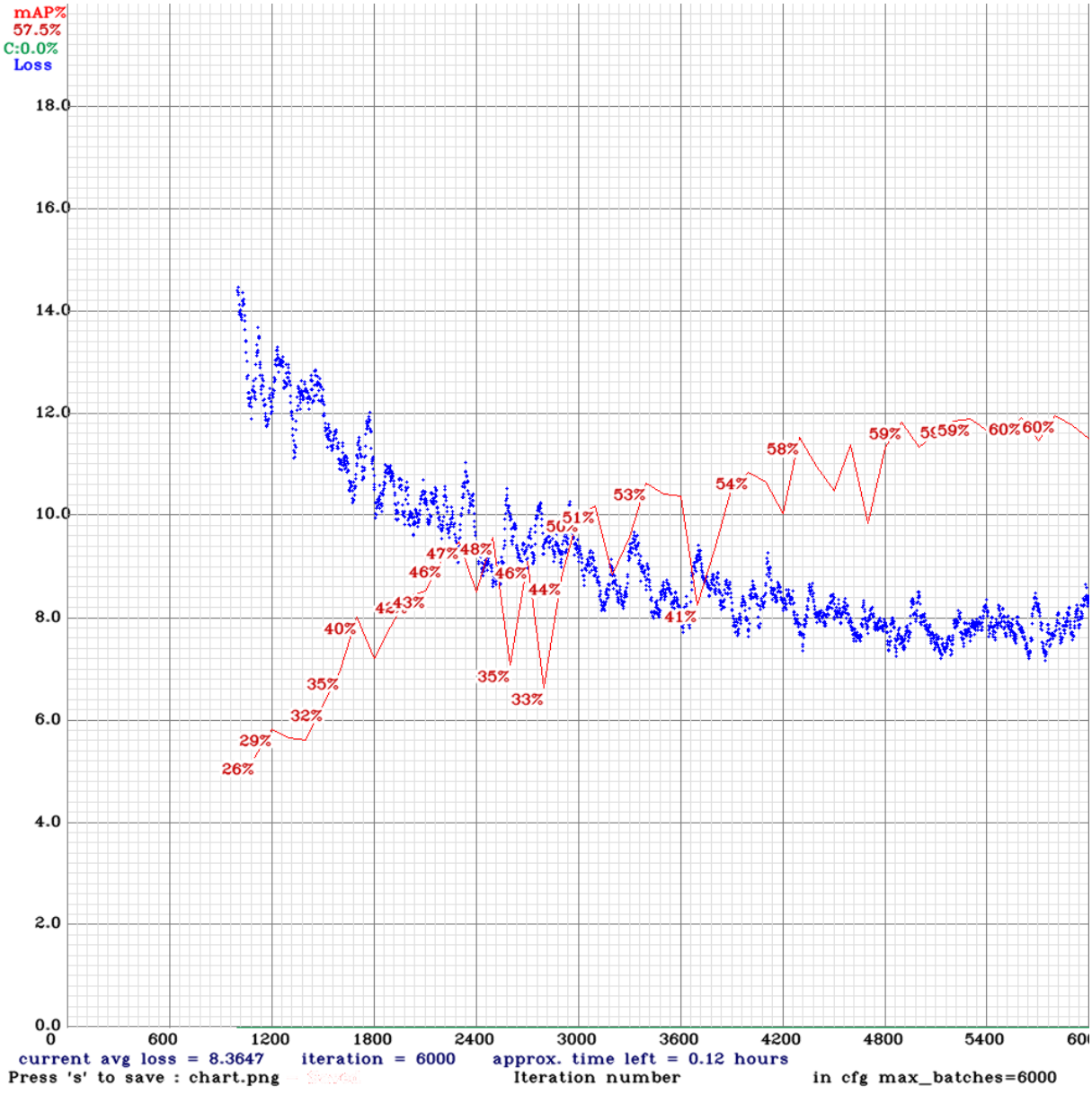

Figura A.6: Gráfico da perda (em azul) e mAP@0.50 (em vermelho) em função das épocas relativo ao treinamento da versão 1 (treinada por 6.000 épocas) do modelo de produção da YOLOv4 utilizando pesos pré-treinados sobre a base de dados MS COCO.

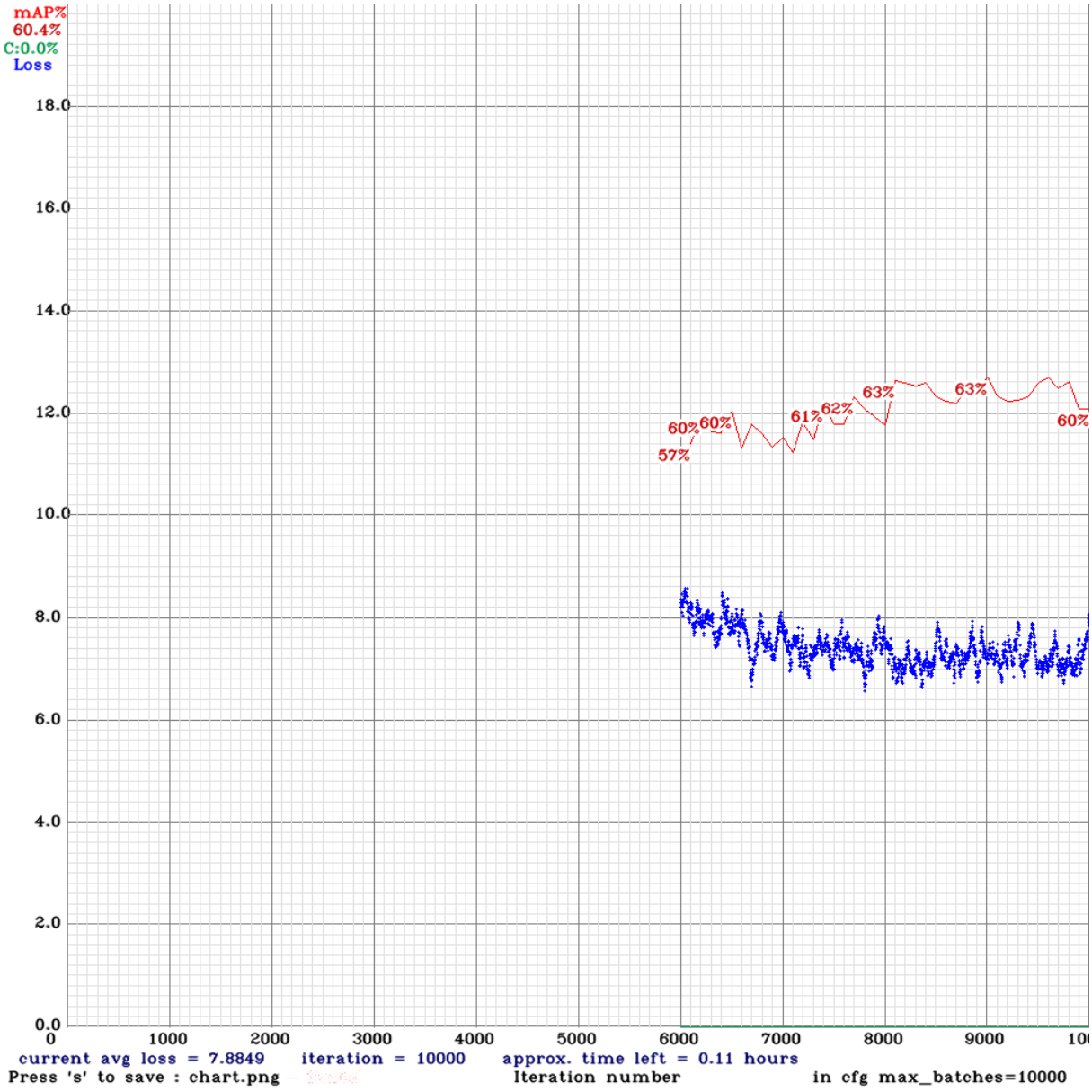

Figura A.7: Gráfico da perda (em azul) e mAP@0.50 (em vermelho) em função das épocas relativo ao treinamento da versão 1 (treinada por um total de 10.000 épocas) do modelo de produção da YOLOv4 utilizando pesos pré-treinados sobre a base de dados MS COCO.

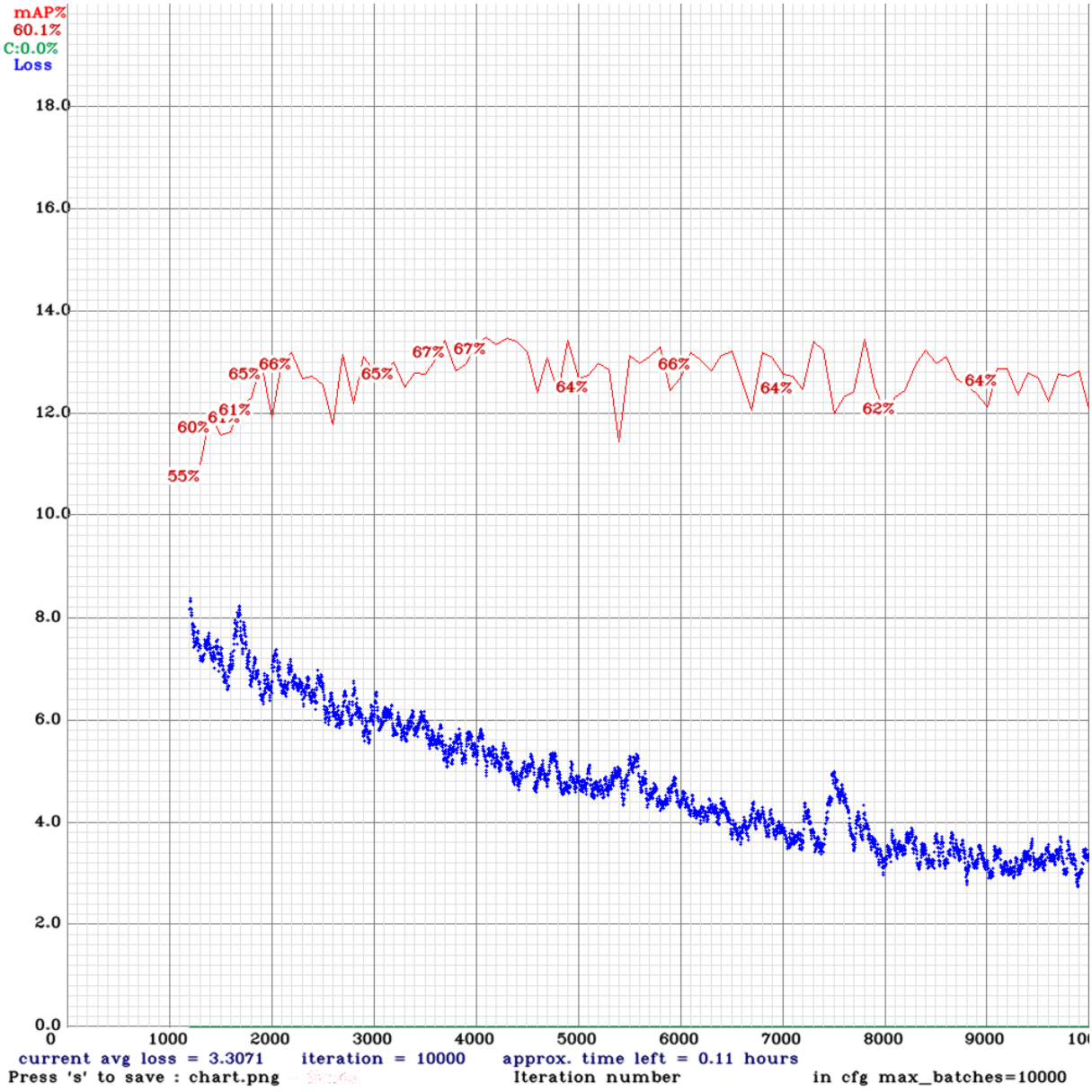

Figura A.8: Gráfico da perda (em azul) e mAP@0.50 (em vermelho) em função das épocas relativo ao treinamento da versão 2 do modelo de produção da YOLOv4 utilizando pesos pré-treinados sobre a base de dados MS COCO.

gradiente.

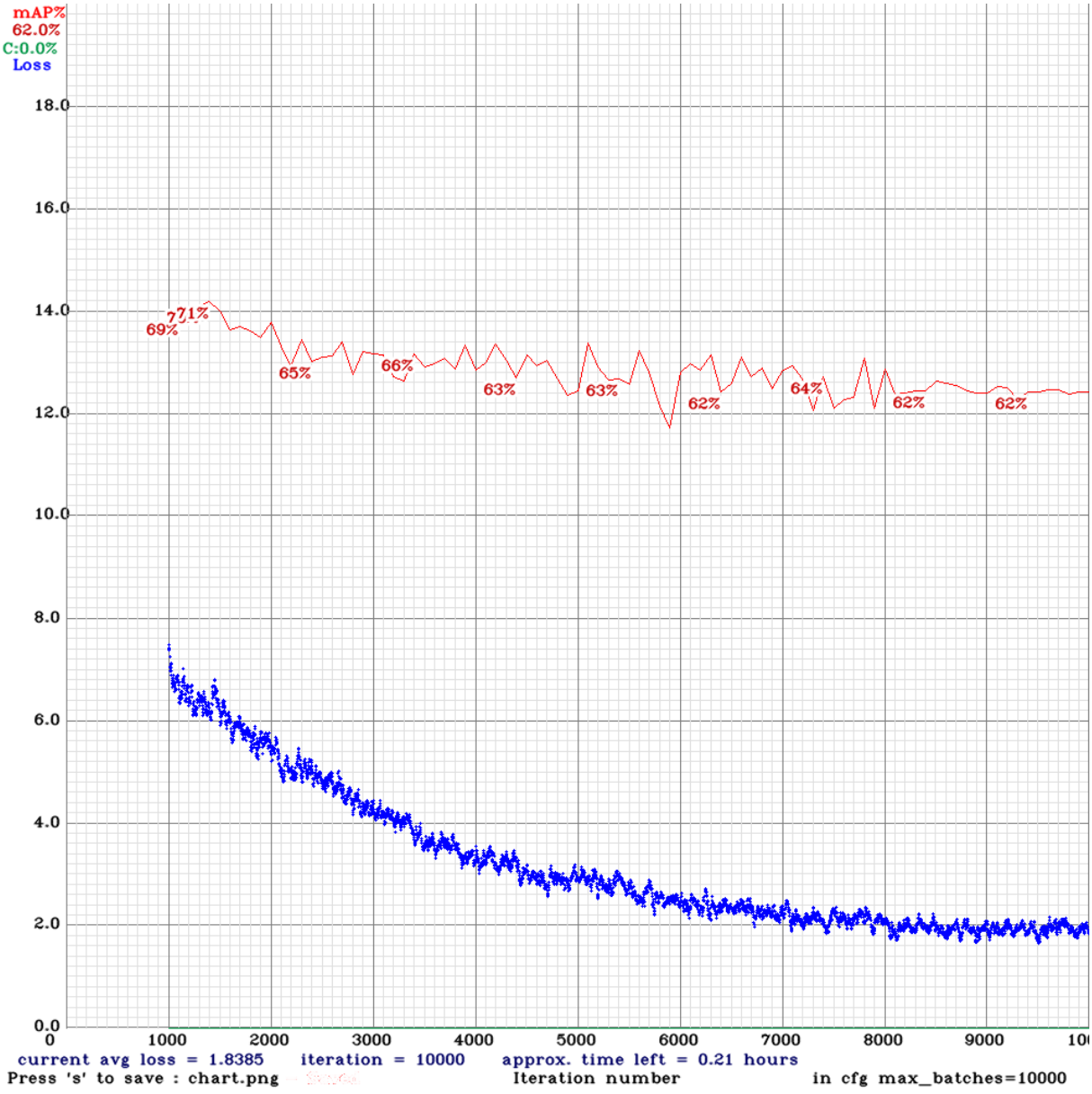

Figura A.9: Gráfico da perda (em azul) e mAP@0.50 (em vermelho) em função das épocas relativo ao treinamento da versão 4 do modelo de produção da YOLOv4 utilizando pesos pré-treinados sobre a base de dados MS COCO.

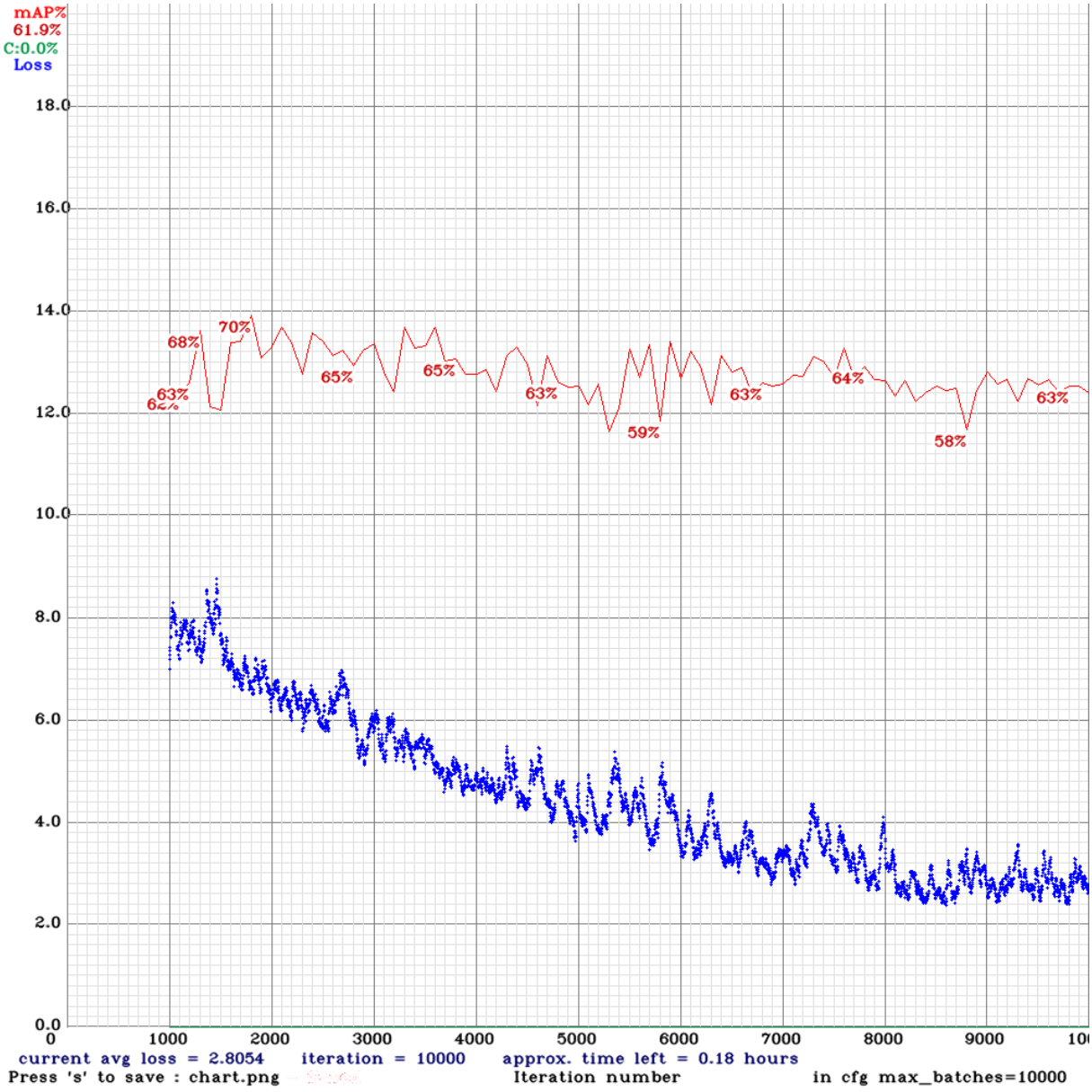

Figura A.10: Gráfico da perda (em azul) e mAP@0.50 (em vermelho) em função das épocas relativo ao treinamento da versão 5 do modelo de produção da YOLOv4 utilizando pesos pré-treinados sobre a base de dados MS COCO.

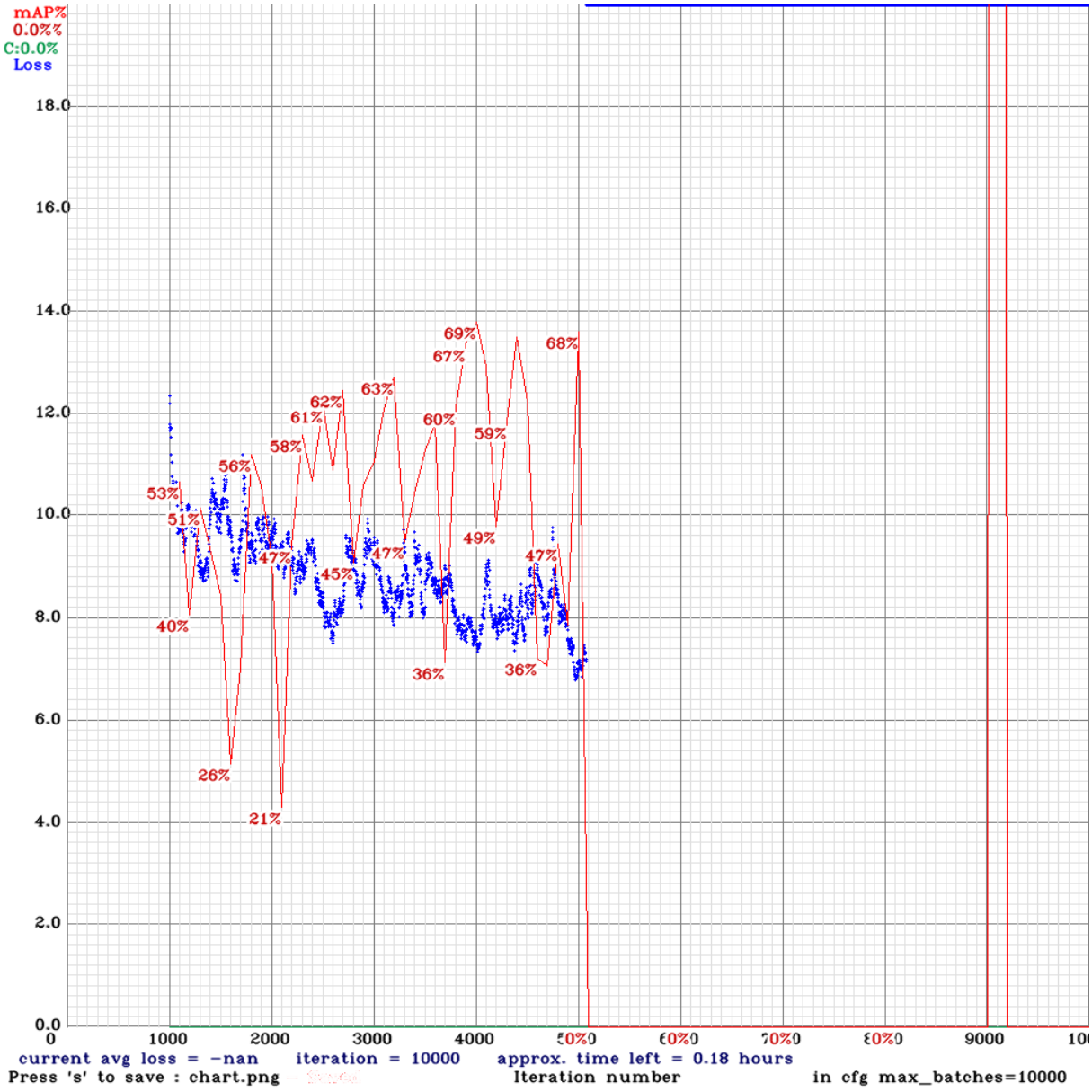

Figura A.11: Gráfico da perda (em azul) e mAP@0.50 (em vermelho) em função das épocas relativo ao treinamento da versão 7 do modelo de produção da YOLOv4 utilizando pesos pré-treinados sobre a base de dados MS COCO.

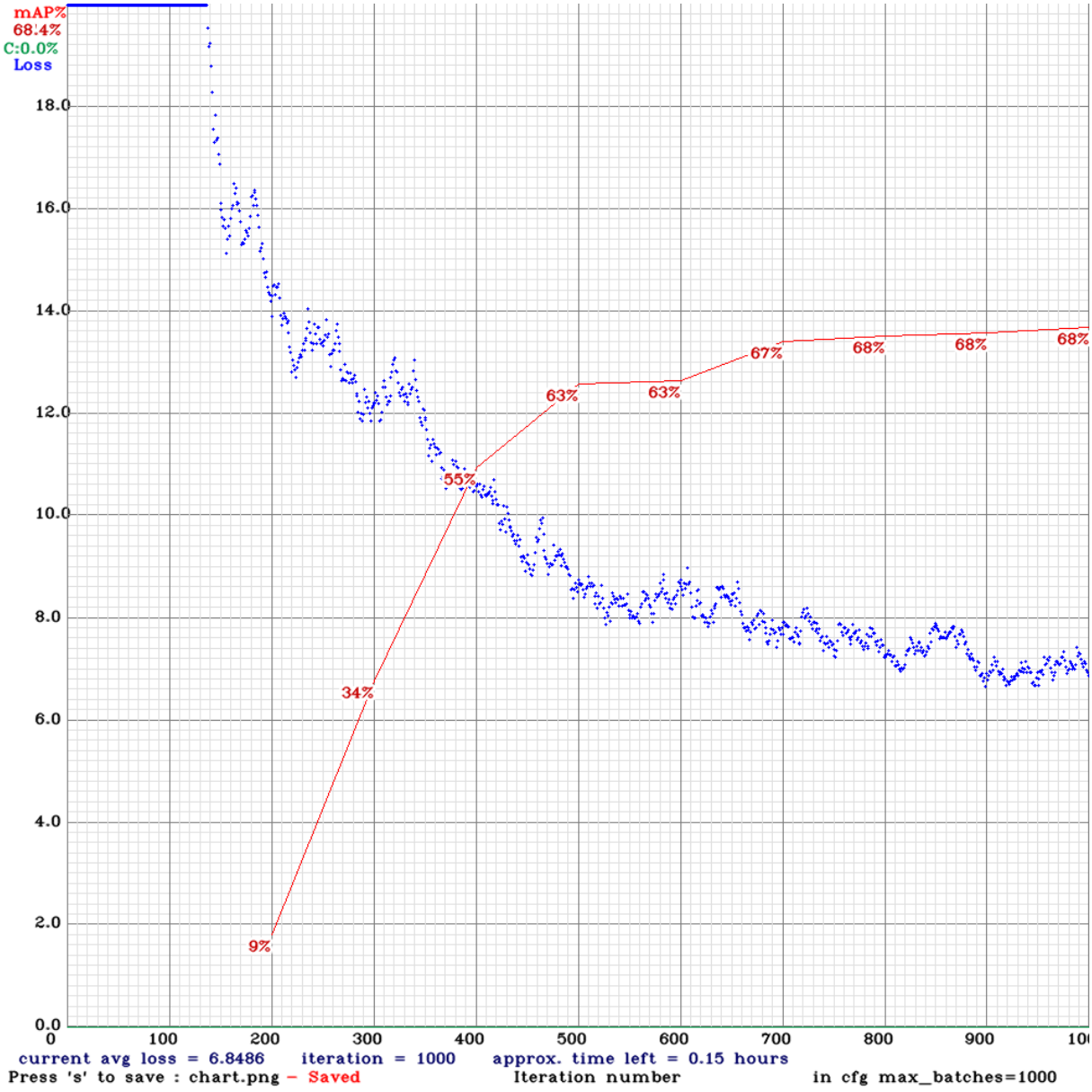

Figura A.12: Gráfico da perda (em azul) e mAP@0.50 (em vermelho) em função das épocas relativo ao treinamento da versão 8 do modelo de produção da YOLOv4 utilizando pesos pré-treinados sobre a base de dados MS COCO.

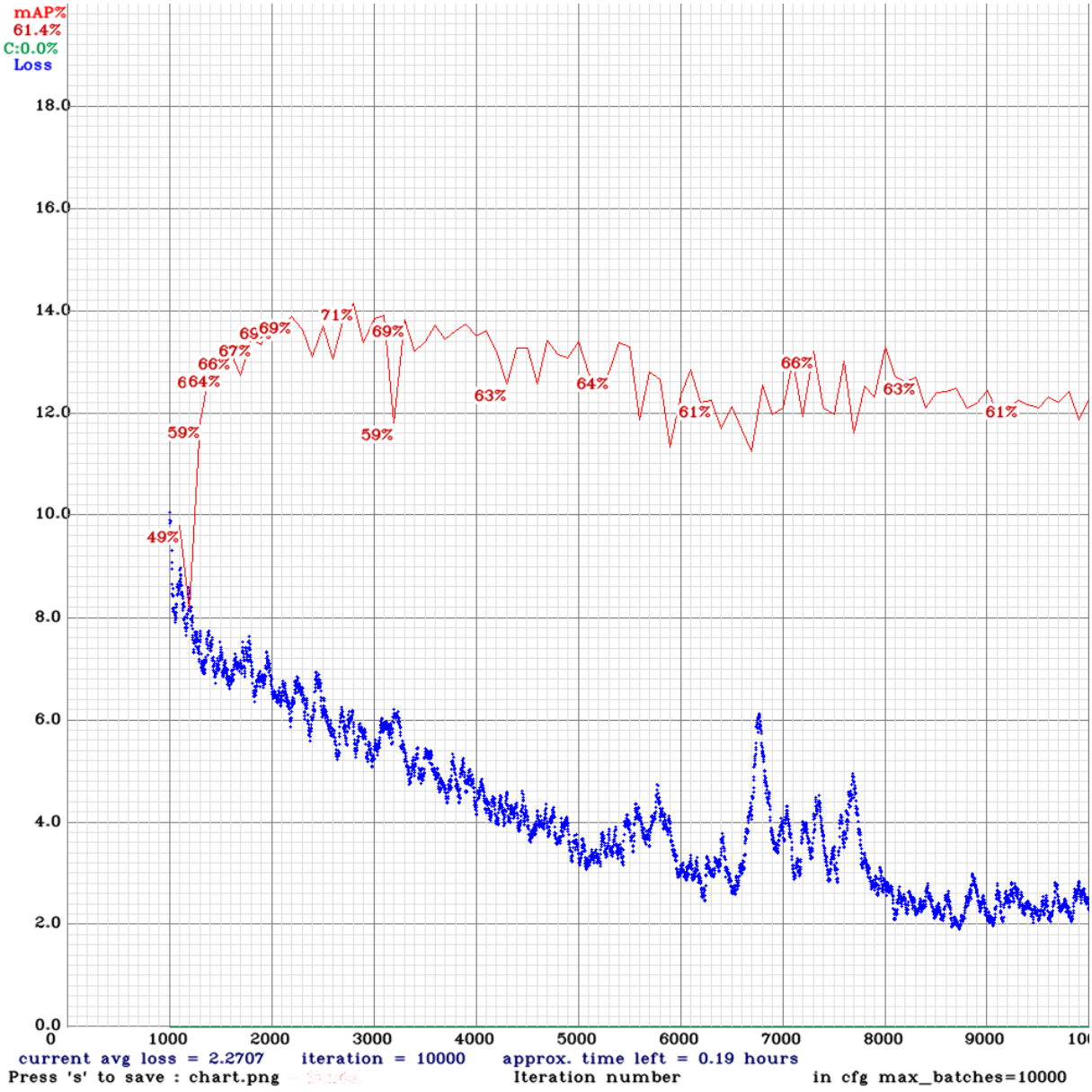

Figura A.13: Gráfico da perda (em azul) e mAP@0.50 (em vermelho) em função das épocas relativo ao treinamento da versão 9 do modelo de produção da YOLOv4 utilizando pesos pré-treinados sobre a base de dados MS COCO.

A.3 Gráficos relativos ao treinamento do modelo utilizando os pesos pré-treinados do sistema de *auto labeling*

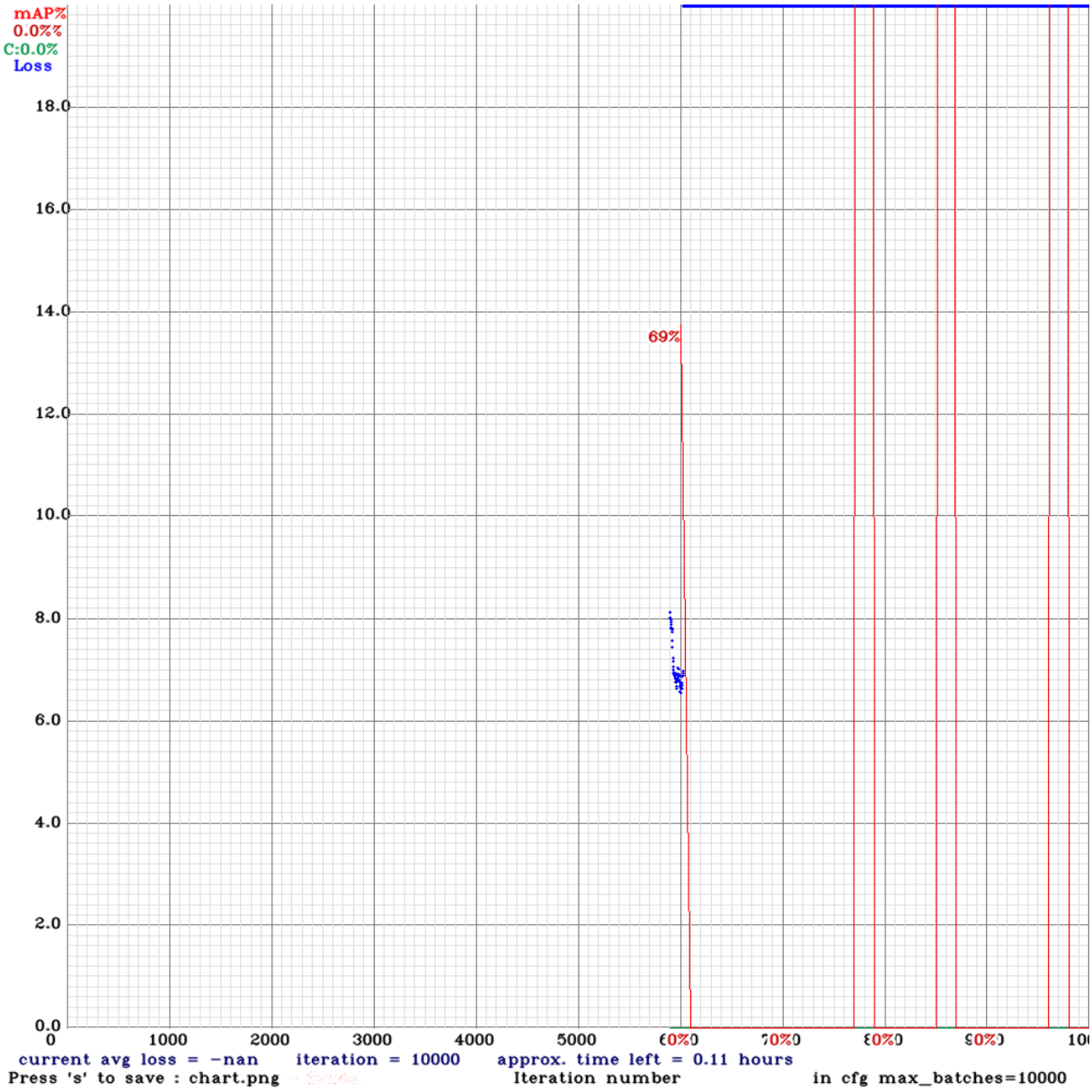

Figura A.14: Gráfico da perda (em azul) e mAP@0.50 (em vermelho) em função das épocas relativo ao treinamento da versão AL1 do modelo de produção da YOLOv4 utilizando pesos pré-treinados do sistema de *auto labeling*.

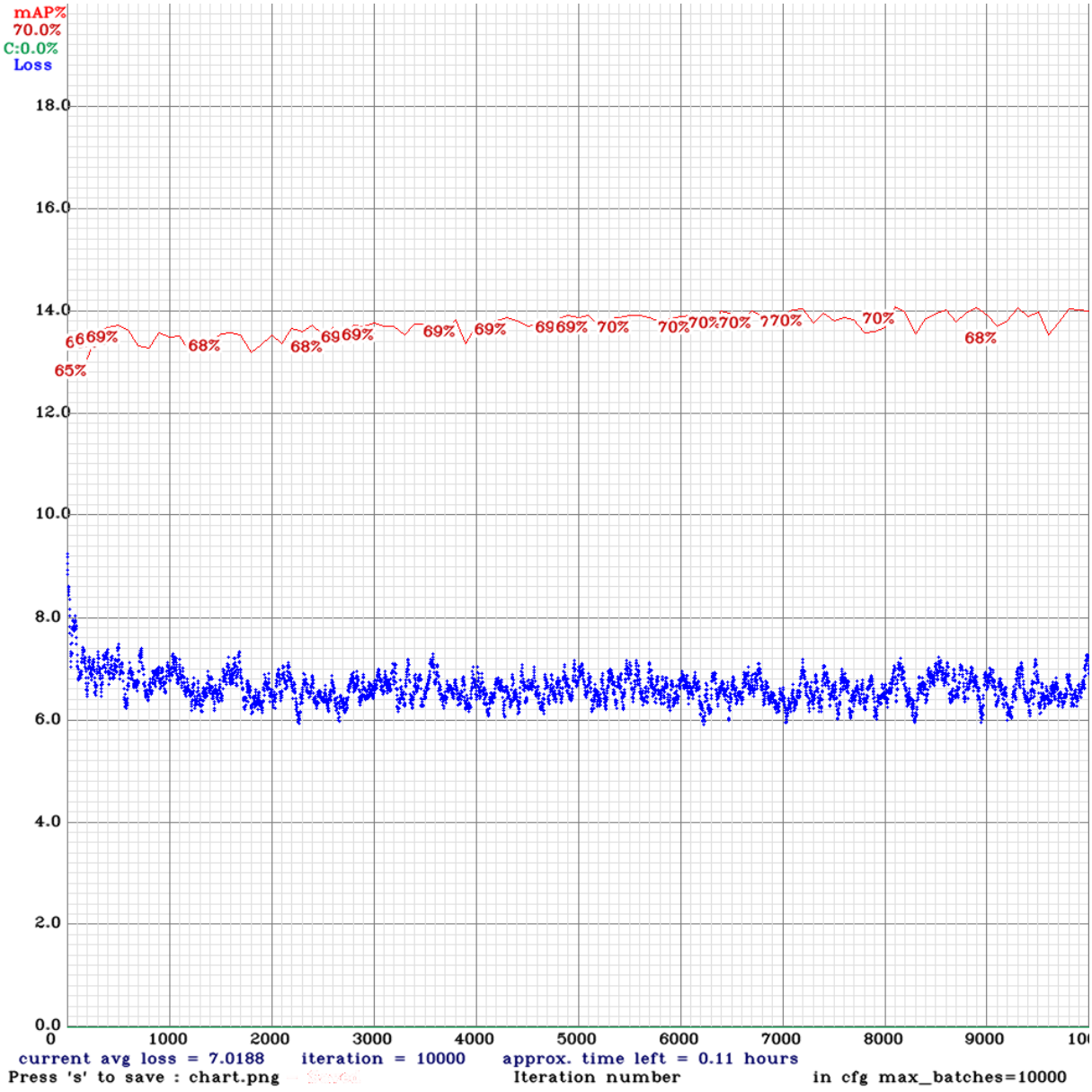

Figura A.15: Gráfico da perda (em azul) e mAP@0.50 (em vermelho) em função das épocas relativo ao treinamento da versão AL2 do modelo de produção da YOLOv4 utilizando pesos pré-treinados do sistema de *auto labeling*.

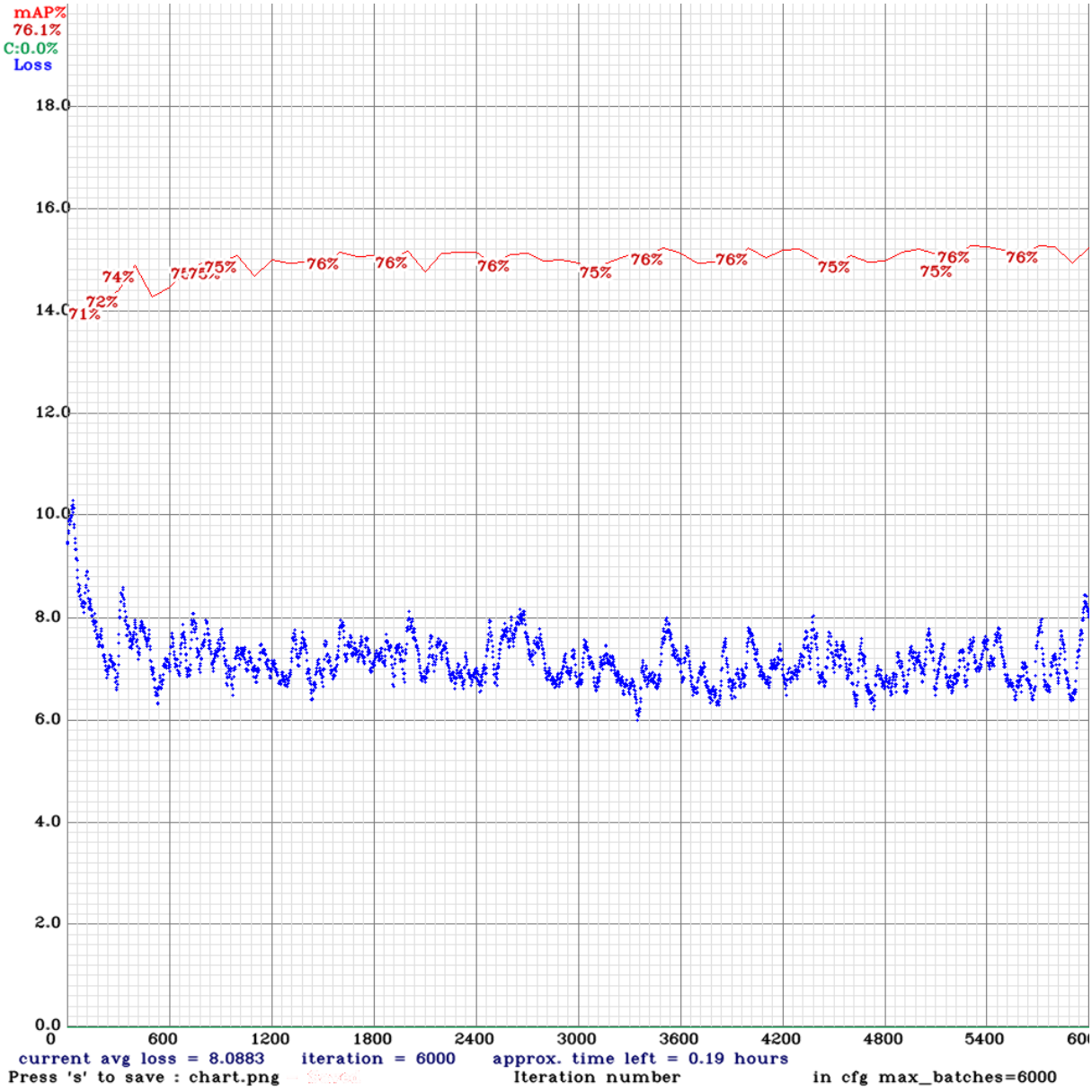

Figura A.16: Gráfico da perda (em azul) e mAP@0.50 (em vermelho) em função das épocas relativo ao treinamento da versão AL3 do modelo de produção da YOLOv4 utilizando pesos pré-treinados do sistema de *auto labeling*.

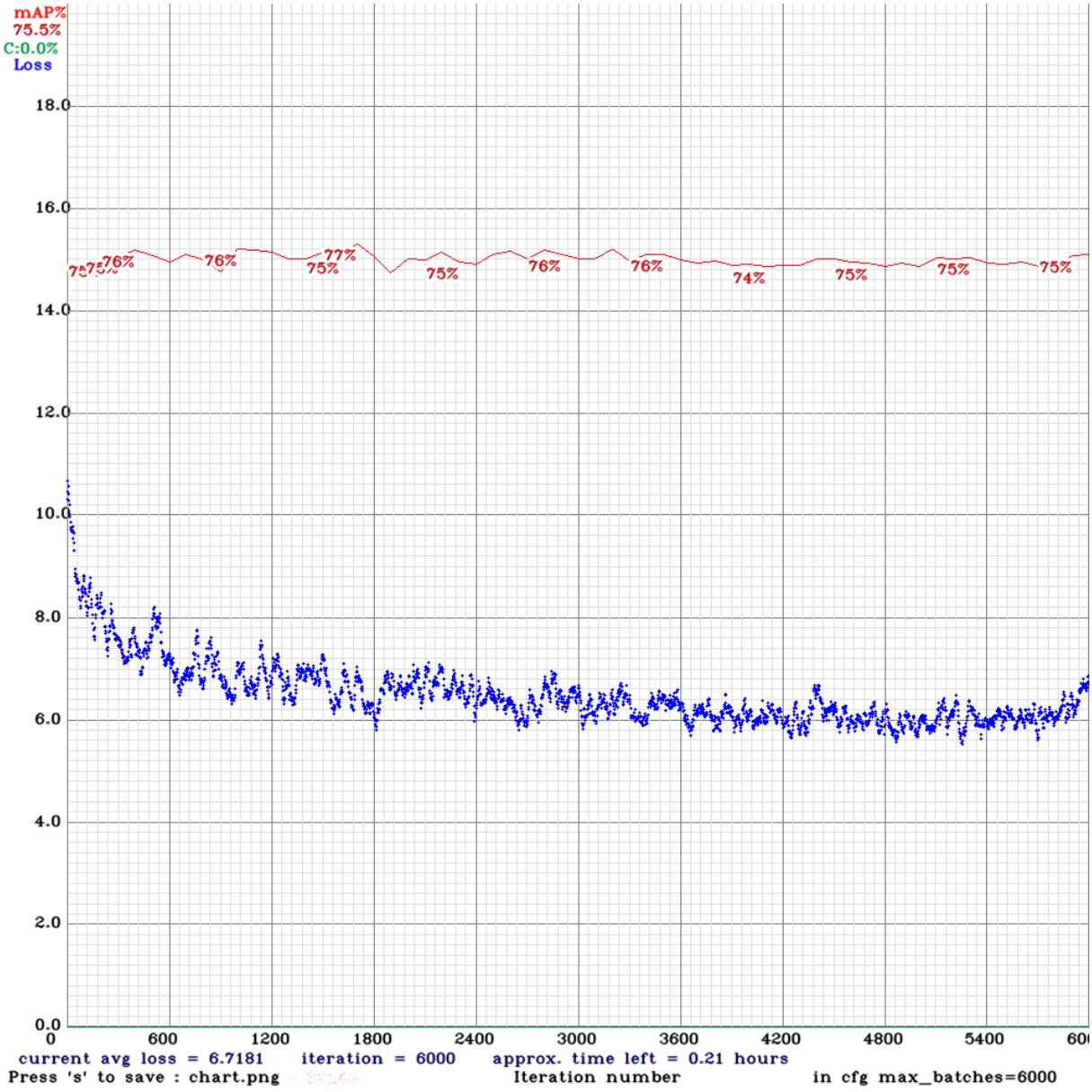

Figura A.17: Gráfico da perda (em azul) e mAP@0.50 (em vermelho) em função das épocas relativo ao treinamento da versão AL4 do modelo de produção da YOLOv4 utilizando pesos pré-treinados do sistema de *auto labeling*.

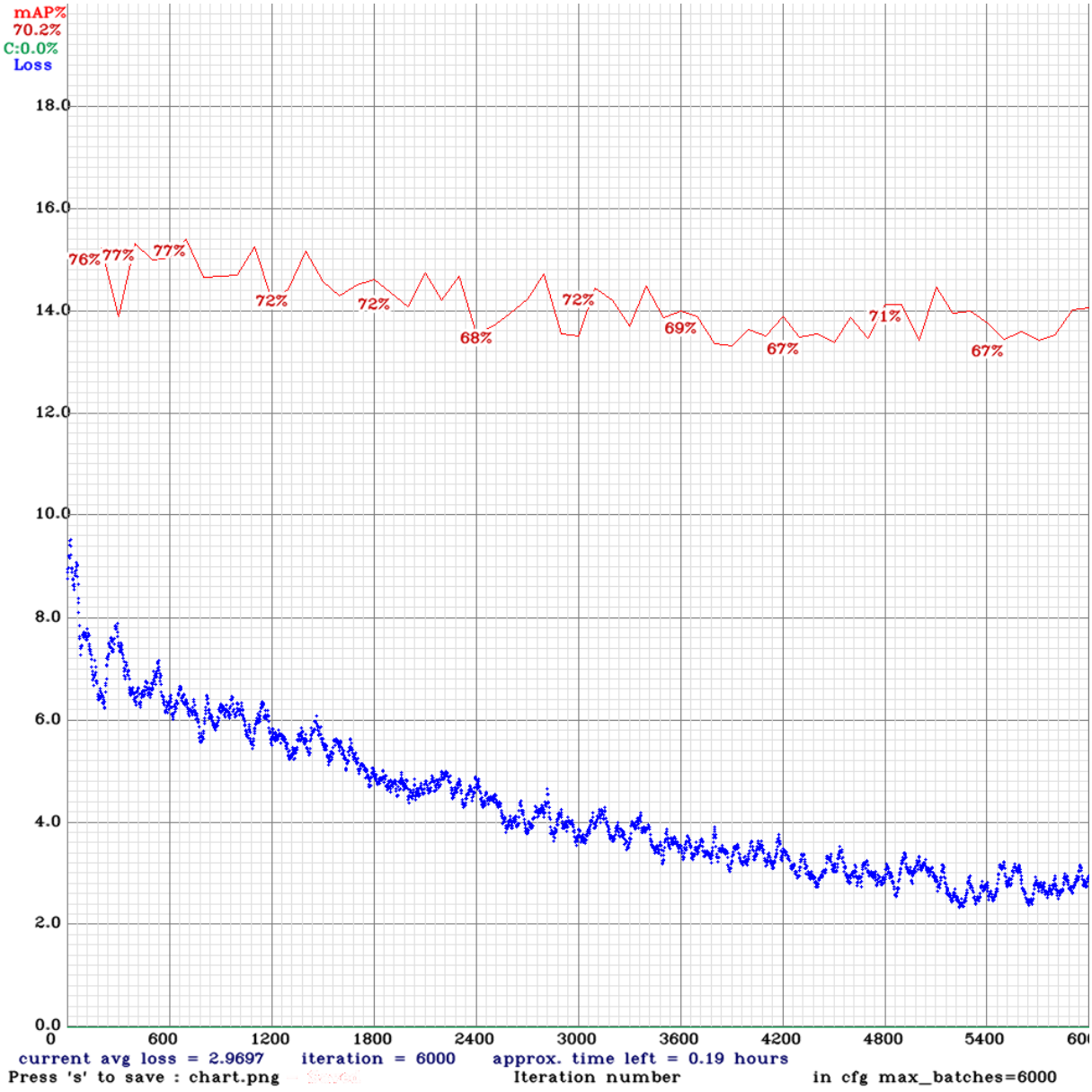

Figura A.18: Gráfico da perda (em azul) e mAP@0.50 (em vermelho) em função das épocas relativo ao treinamento da versão AL6 do modelo de produção da YOLOv4 utilizando pesos pré-treinados do sistema de *auto labeling*.

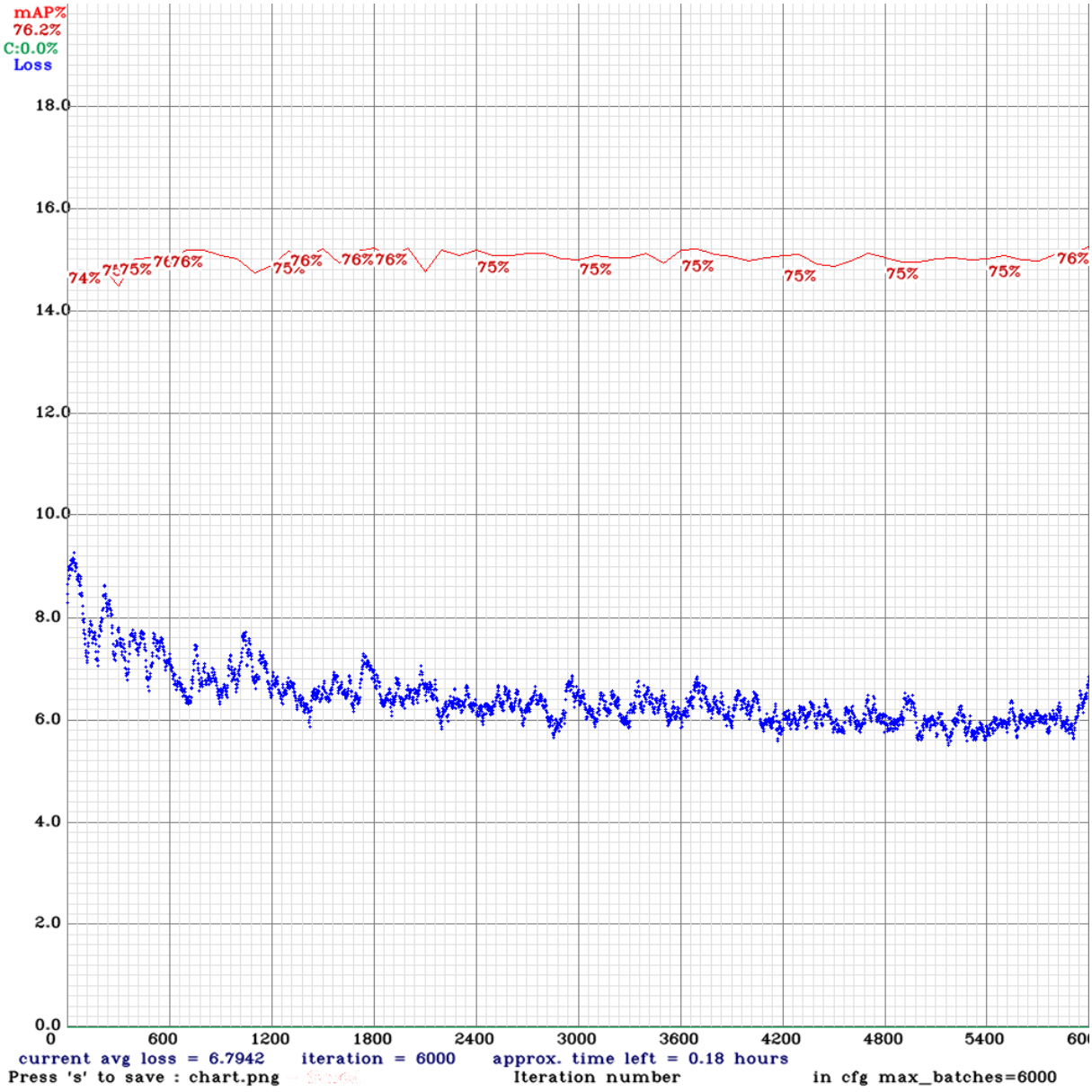

Figura A.19: Gráfico da perda (em azul) e mAP@0.50 (em vermelho) em função das épocas relativo ao treinamento da versão AL7 do modelo de produção da YOLOv4 utilizando pesos pré-treinados do sistema de *auto labeling*.

A.4 Gráficos relativos ao teste de validação cruzada *k-fold*

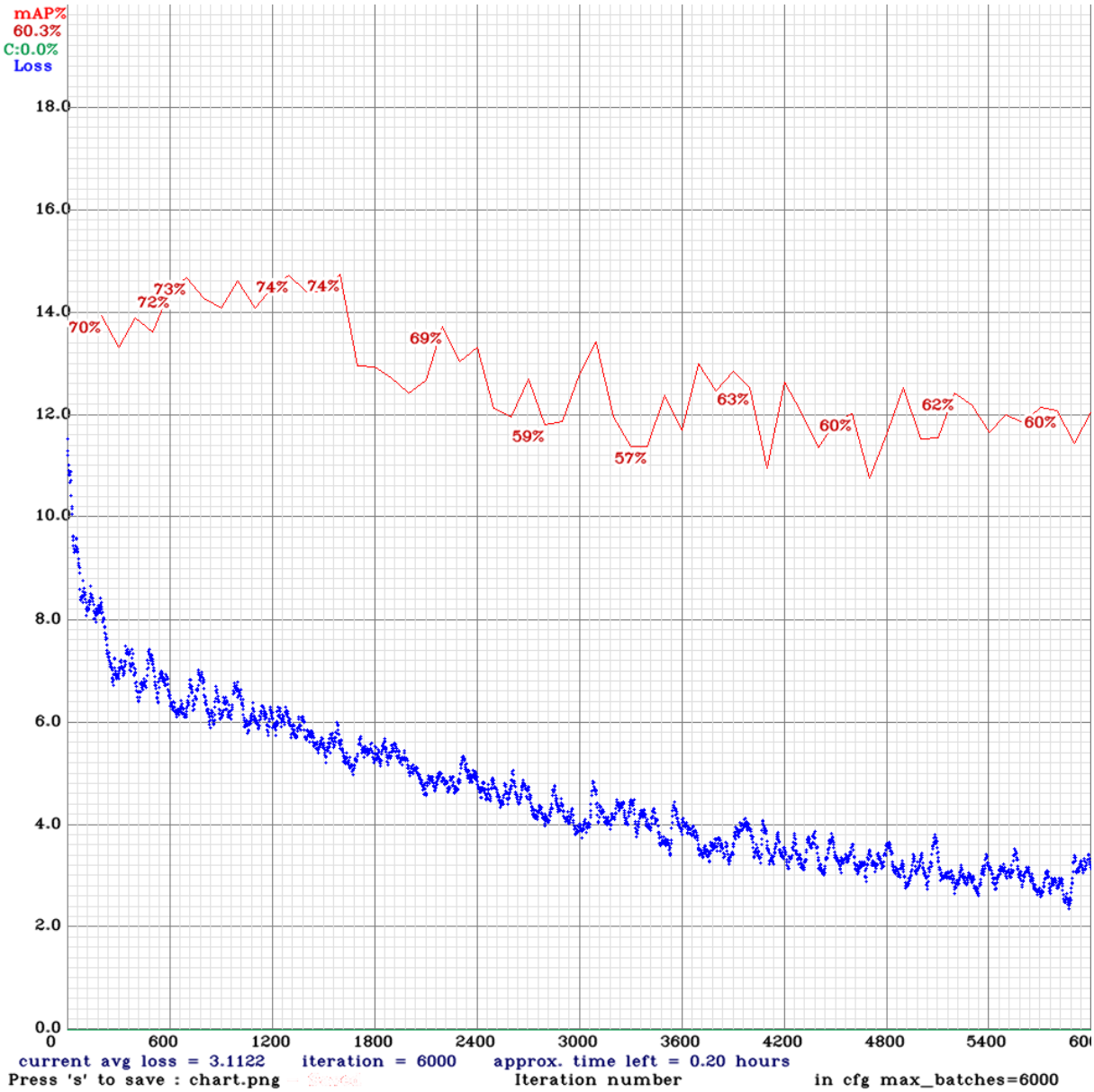

Figura A.20: Gráfico da perda (em azul) e mAP@0.50 (em vermelho) em função das épocas relativo ao treinamento do *fold* 1 da validação cruzada *k-fold* da YOLOv4.

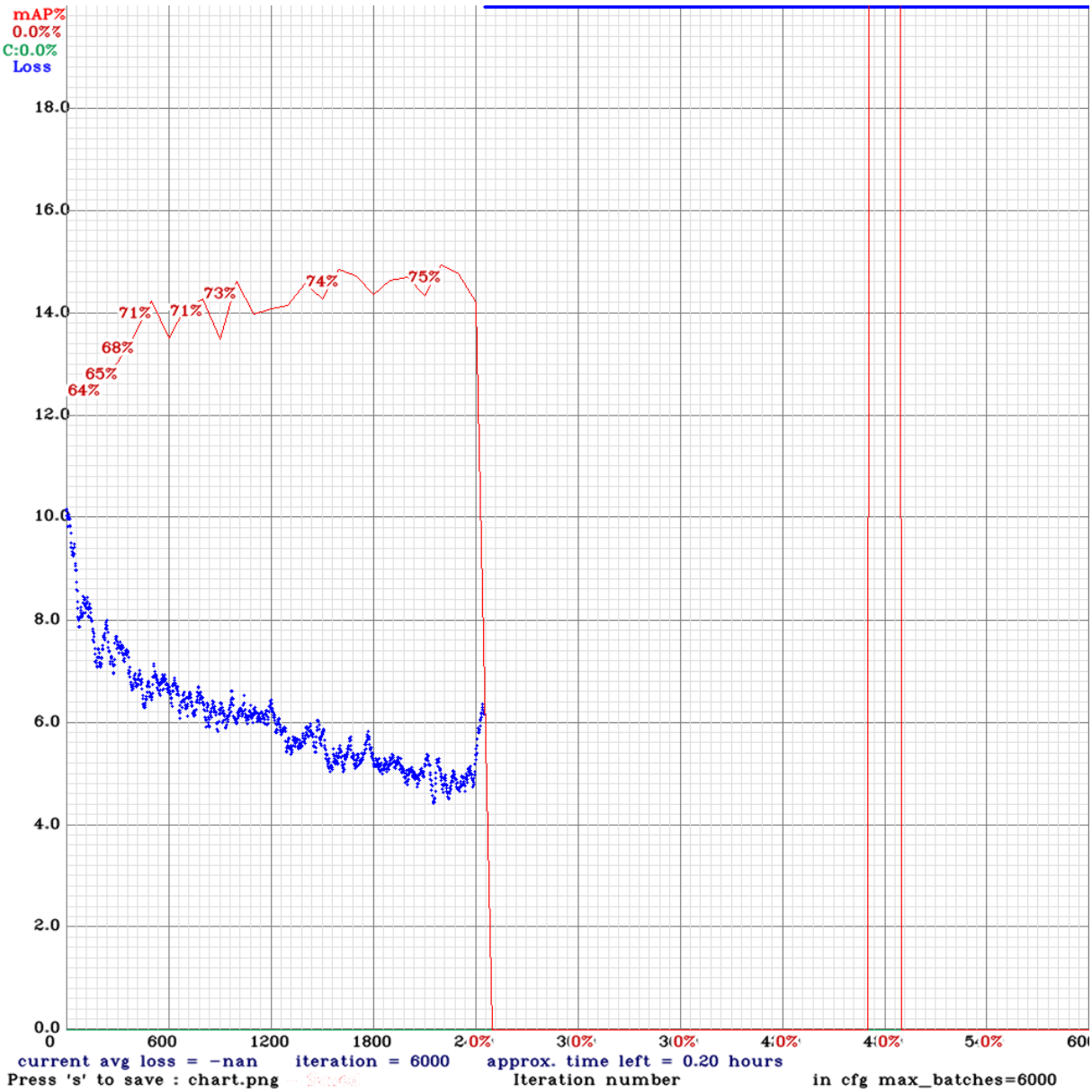

Figura A.21: Gráfico da perda (em azul) e mAP@0.50 (em vermelho) em função das épocas relativo ao treinamento do *fold* 2 da validação cruzada *k-fold* da YOLOv4.

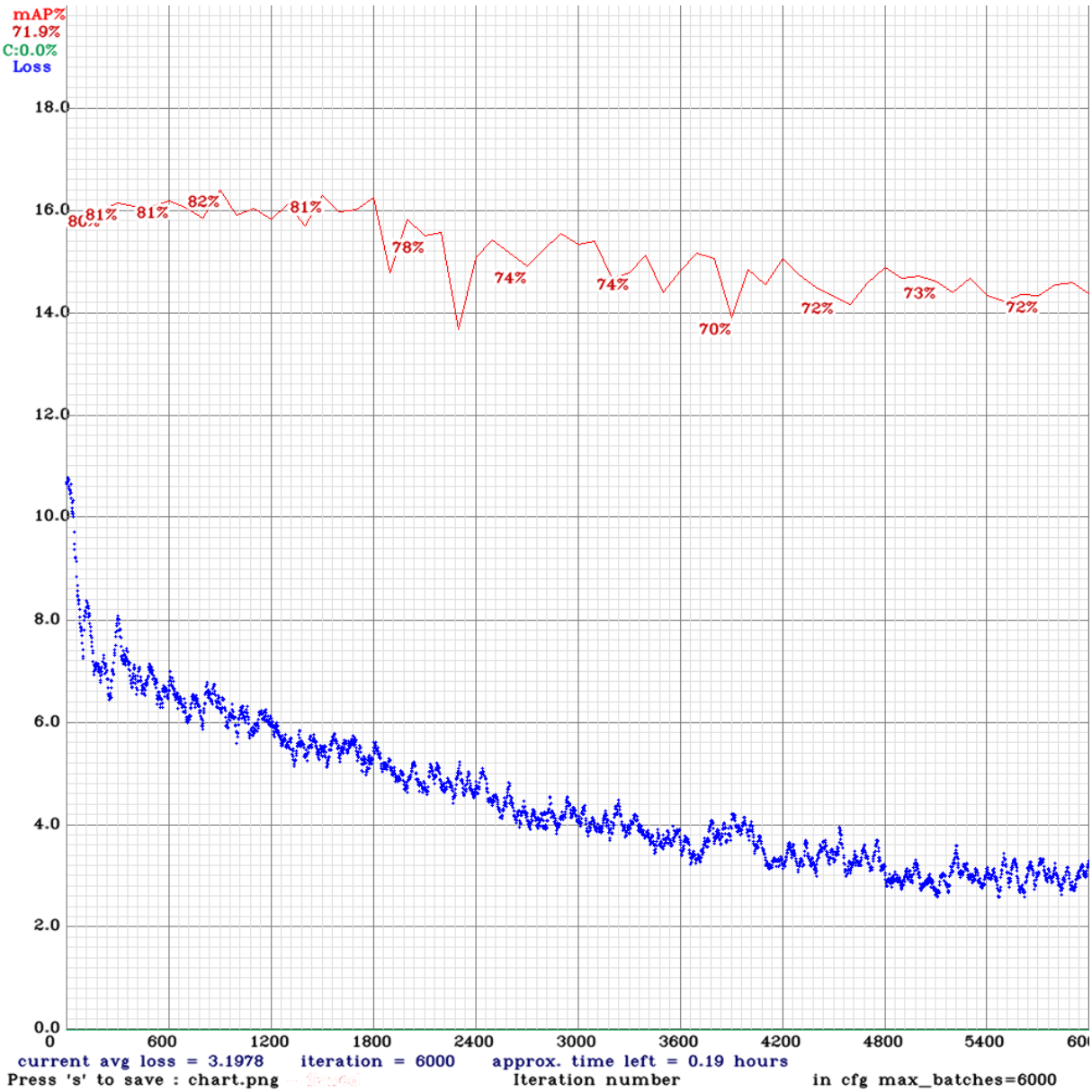

Figura A.22: Gráfico da perda (em azul) e mAP@0.50 (em vermelho) em função das épocas relativo ao treinamento do *fold* 4 da validação cruzada *k-fold* da YOLOv4.

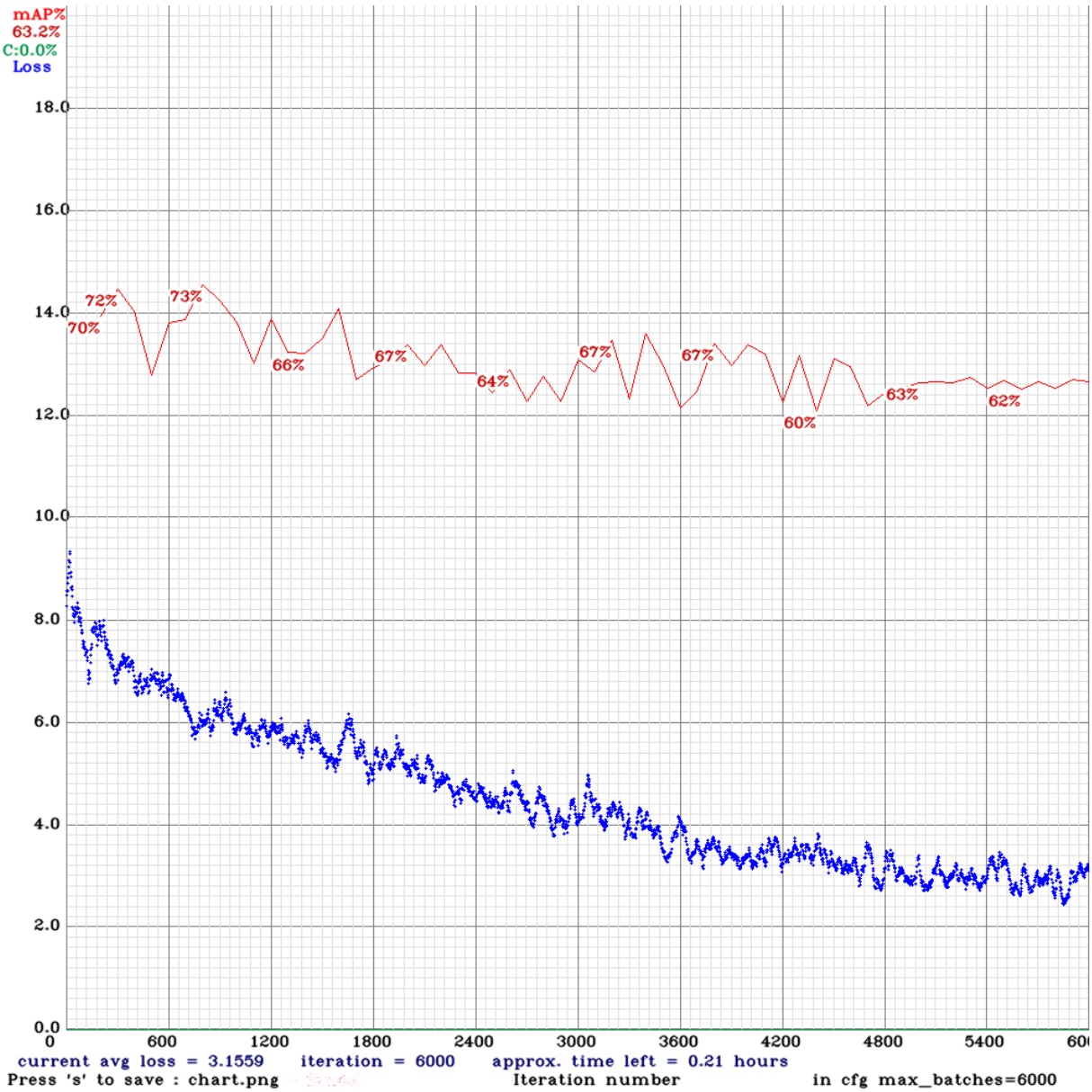

Figura A.23: Gráfico da perda (em azul) e mAP@0.50 (em vermelho) em função das épocas relativo ao treinamento do *fold* 5 da validação cruzada *k-fold* da YOLOv4.

Apêndice B

Gráficos de treinamento e validação da YOLOv9

B.1 Gráficos relativos aos treinamentos do modelos

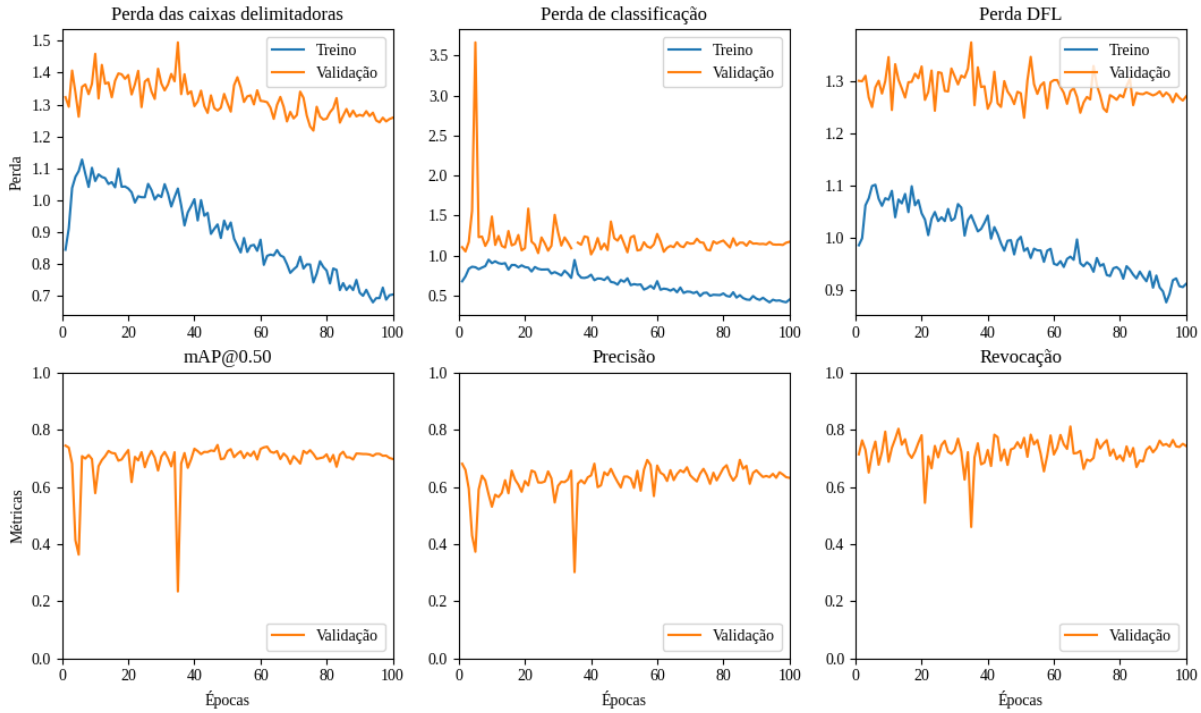

Figura B.1: Gráficos das perdas das caixas delimitadoras, de classificação e DFL; mAP@0.50, precisão e revocação em função das épocas relativo ao treinamento da versão Y9-2 utilizando a YOLOv9.

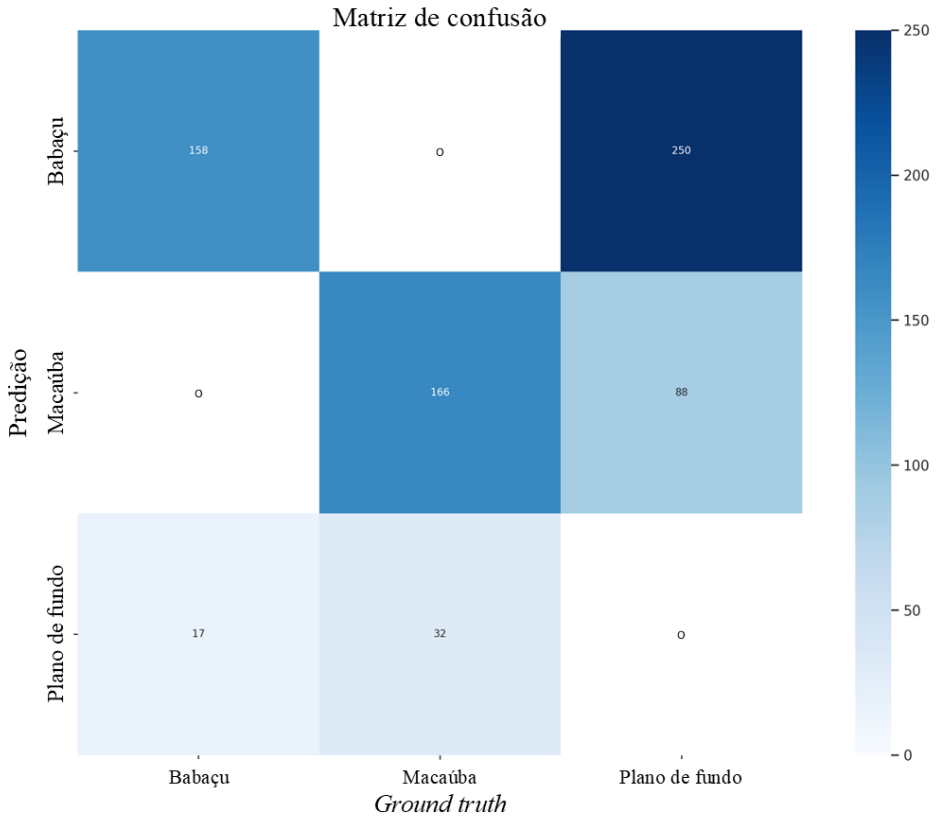

Figura B.2: Matriz de confusão relativa à versão Y9-2 da YOLOv9.

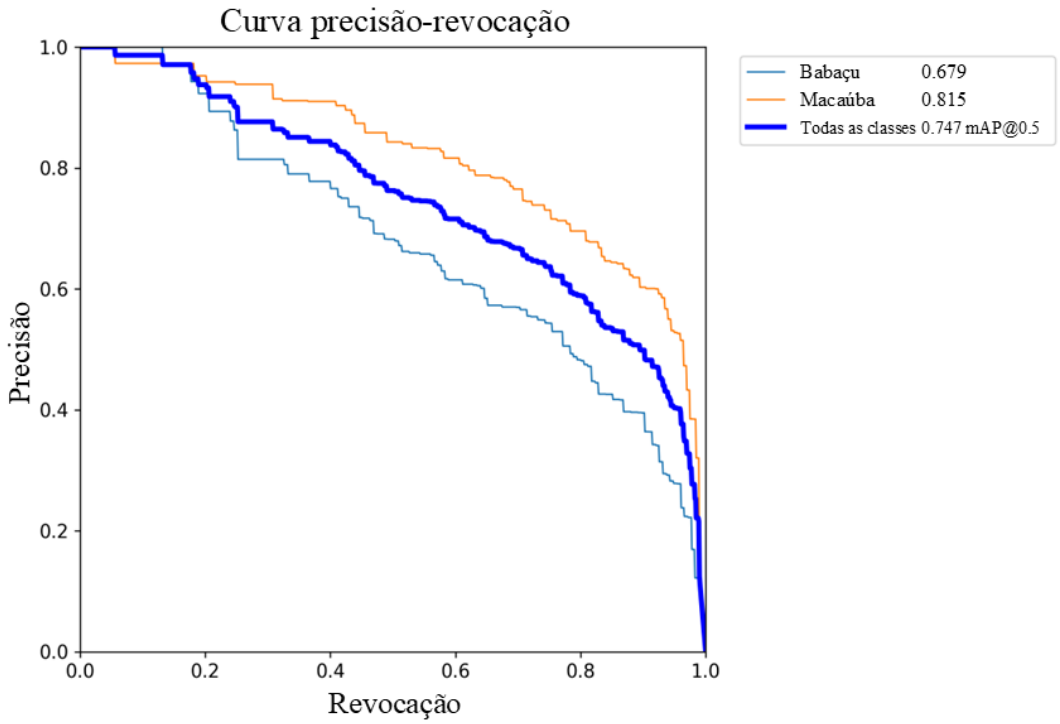

Figura B.3: Curva precisão-revocação relativa à versão Y9-2 da YOLOv9.

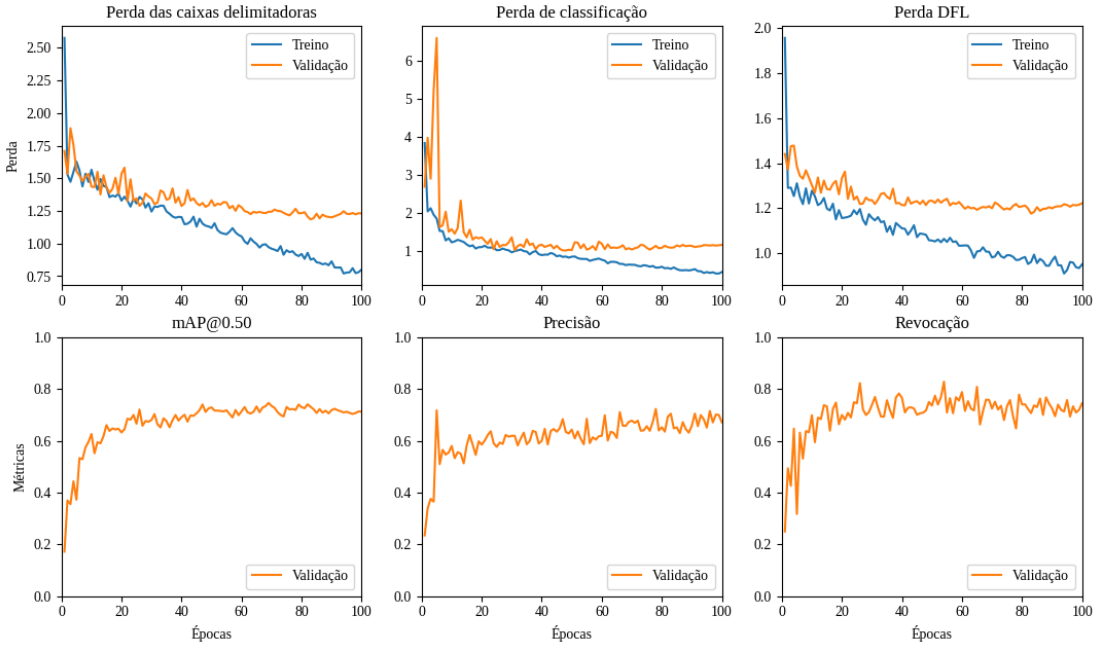

Figura B.4: Gráficos das perdas das caixas delimitadoras, de classificação e DFL; mAP@ 0.50, precisão e revocação em função das épocas relativo ao treinamento da versão Y9-3 utilizando a YOLOv9.

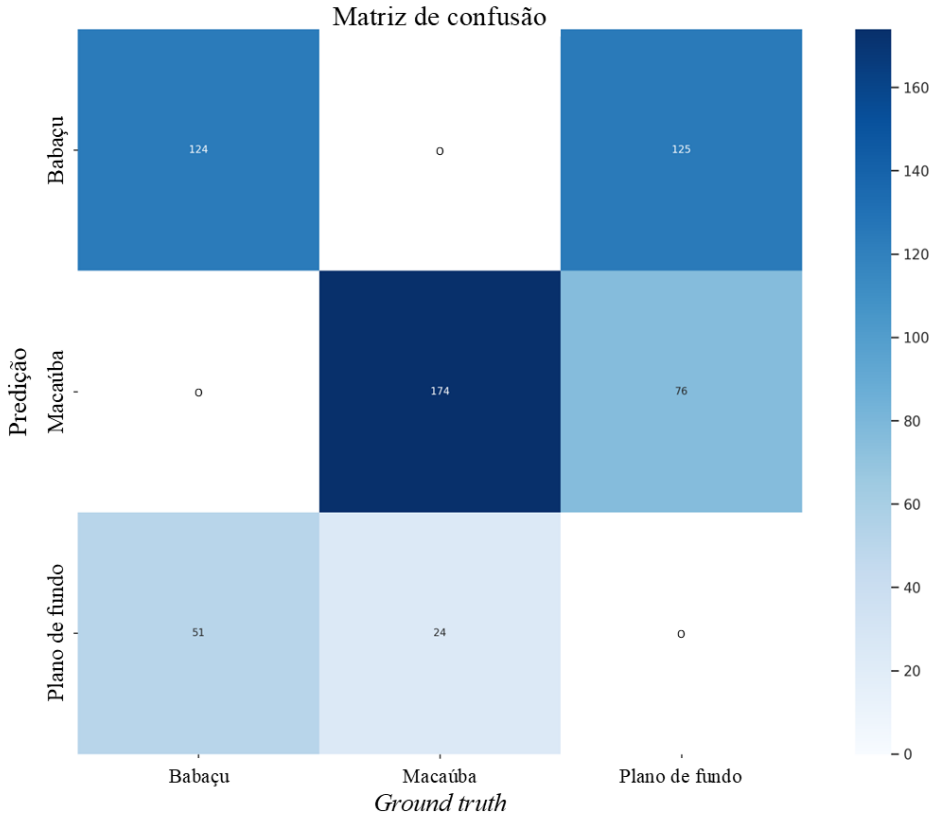

Figura B.5: Matriz de confusão relativa à versão Y9-3 da YOLOv9.

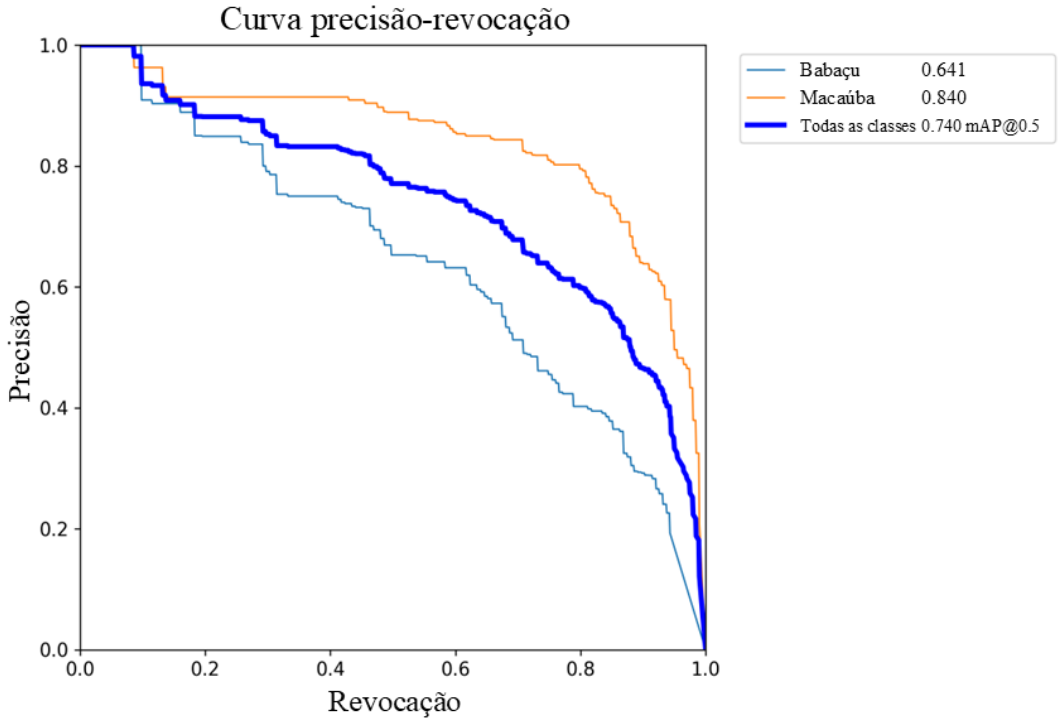

Figura B.6: Curva precisão-revocação relativa à versão Y9-3 da YOLOv9.

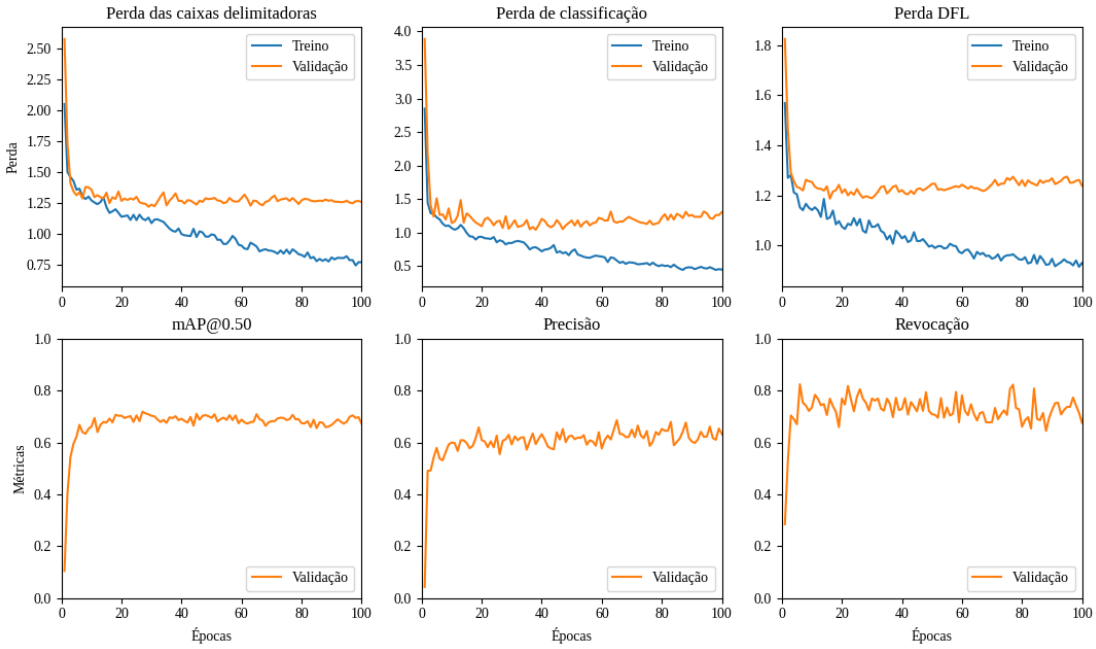

Figura B.7: Gráficos das perdas das caixas delimitadoras, de classificação e DFL; mAP@ 0.50, precisão e revocação em função das épocas relativo ao treinamento da versão Y9-4 utilizando a YOLOv9.

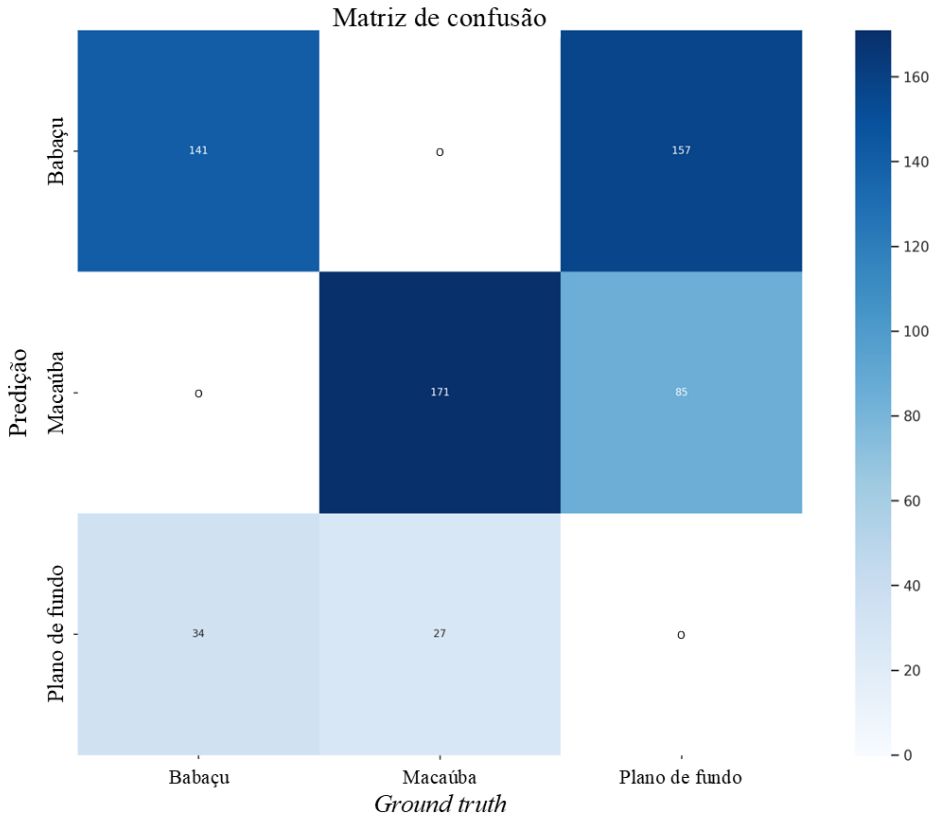

Figura B.8: Matriz de confusão relativa à versão Y9-4 da YOLOv9.

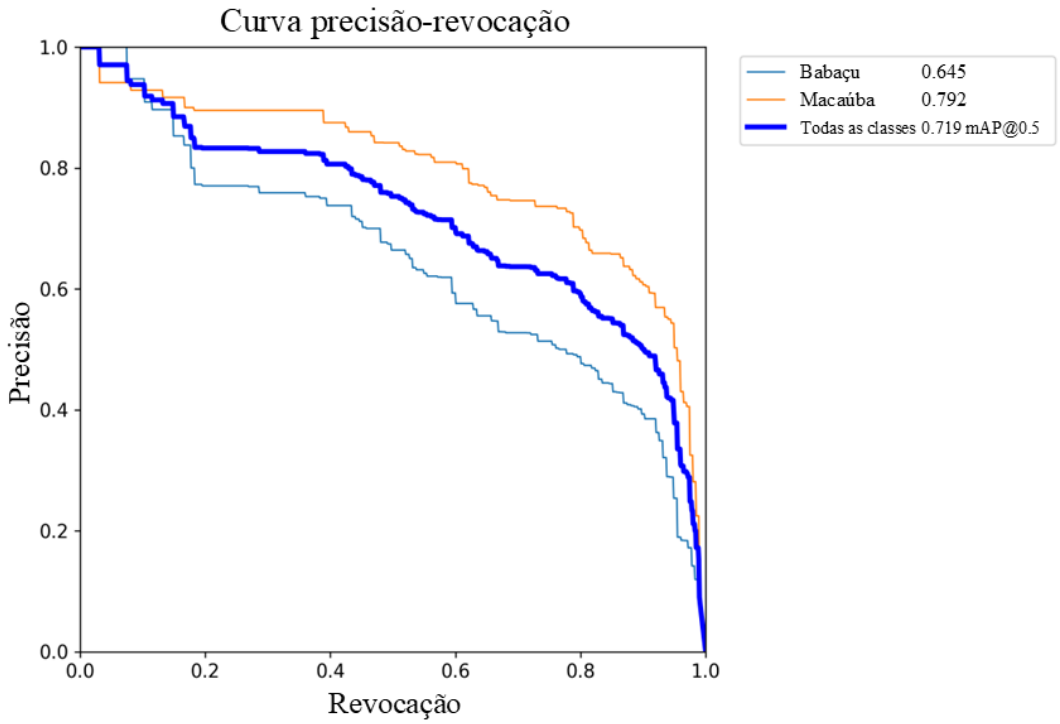

Figura B.9: Curva precisão-revocação relativa à versão Y9-4 da YOLOv9.

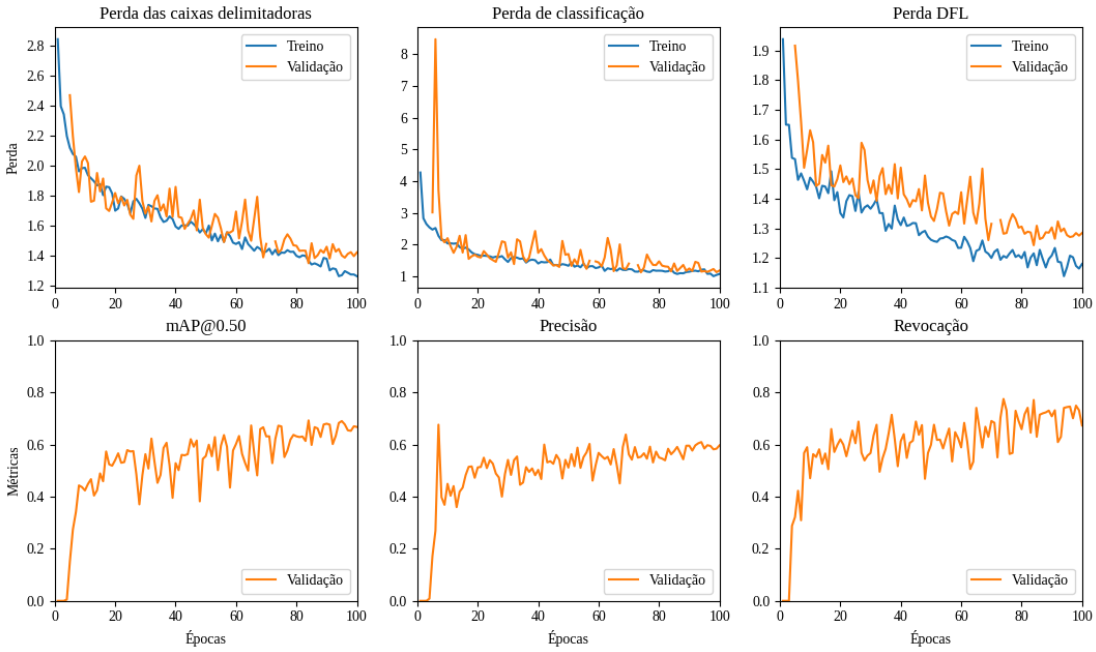

Figura B.10: Gráficos das perdas das caixas delimitadoras, de classificação e DFL; mAP@ 0.50, precisão e revocação em função das épocas relativo ao treinamento da versão Y9-5 utilizando a YOLOv9.

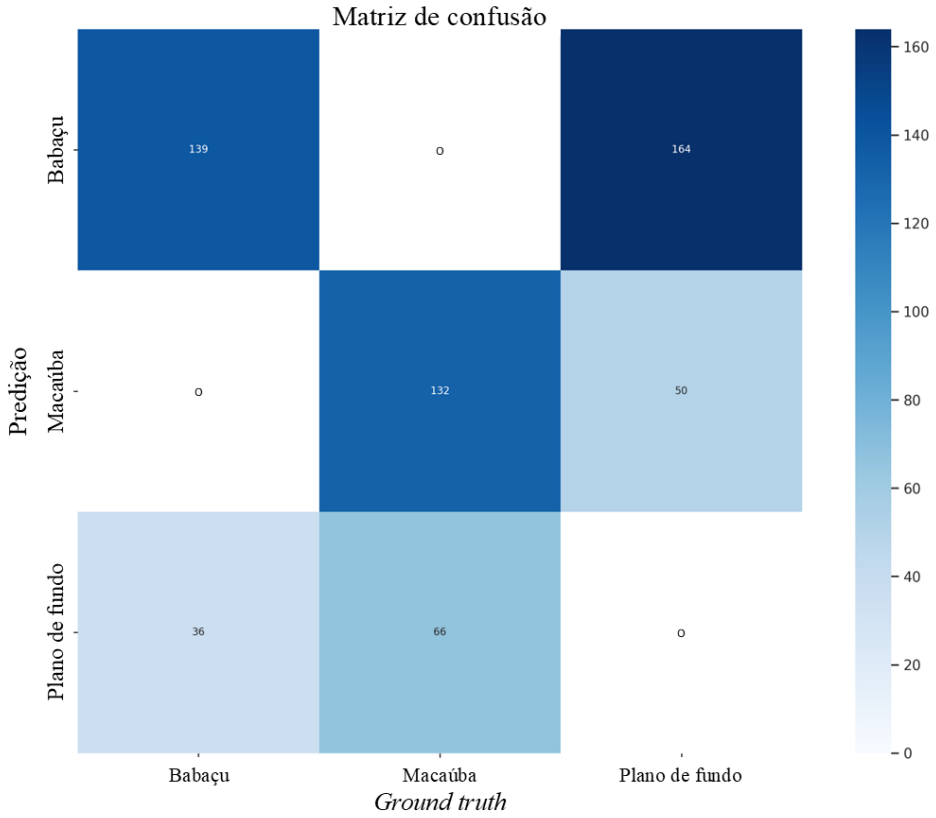

Figura B.11: Matriz de confusão relativa à versão Y9-5 da YOLOv9.

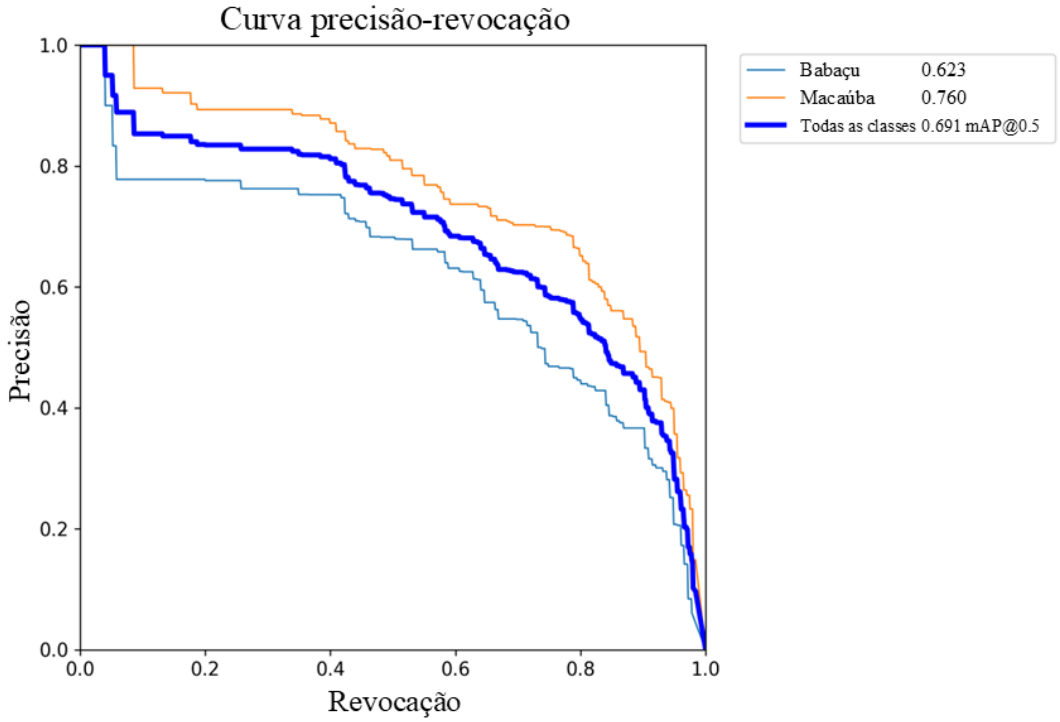

Figura B.12: Curva precisão-revocação relativa à versão Y9-5 da YOLOv9.

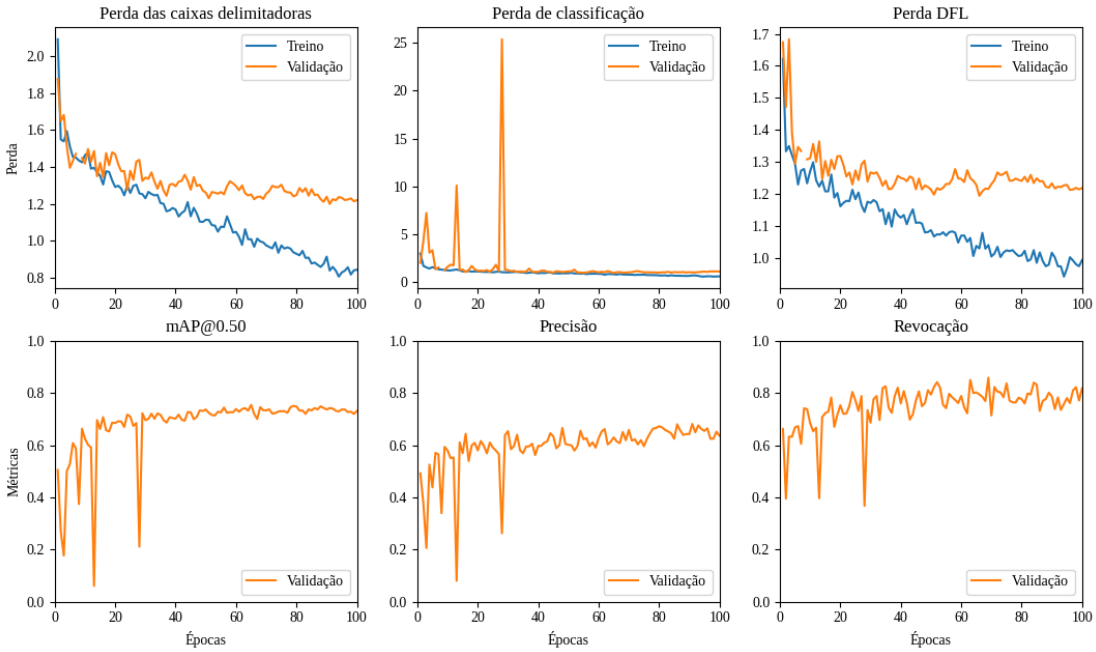

Figura B.13: Gráficos das perdas das caixas delimitadoras, de classificação e DFL; mAP@ 0.50, precisão e revocação em função das épocas relativo ao treinamento da versão Y9-6 utilizando a YOLOv9.

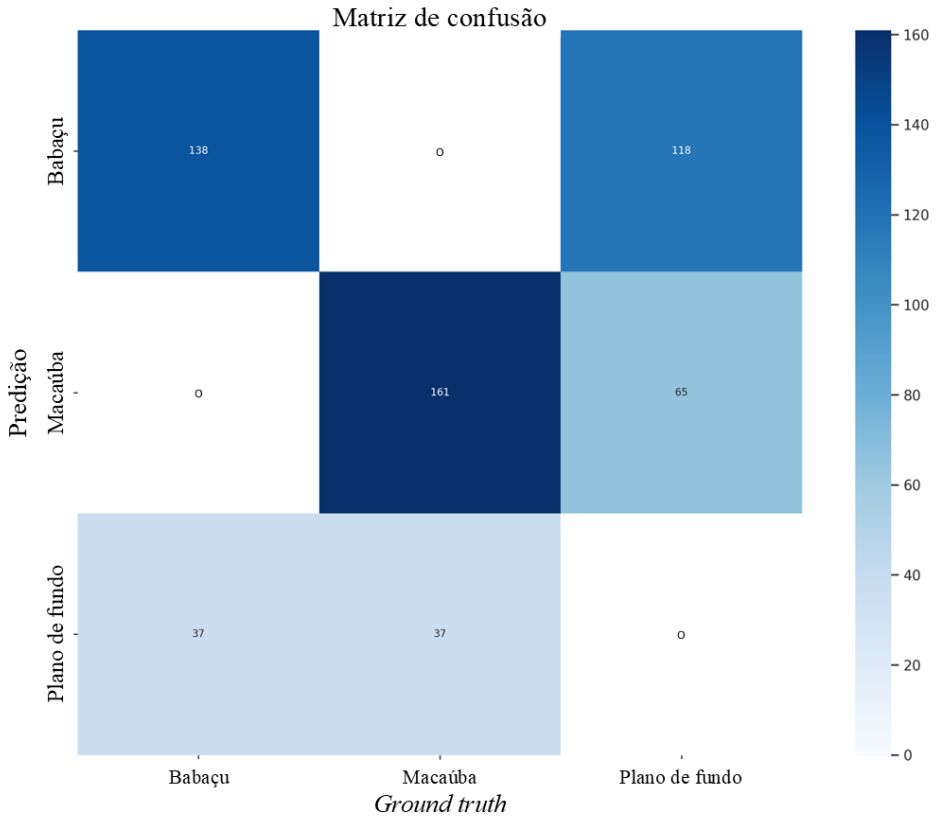

Figura B.14: Matriz de confusão relativa à versão Y9-6 da YOLOv9.

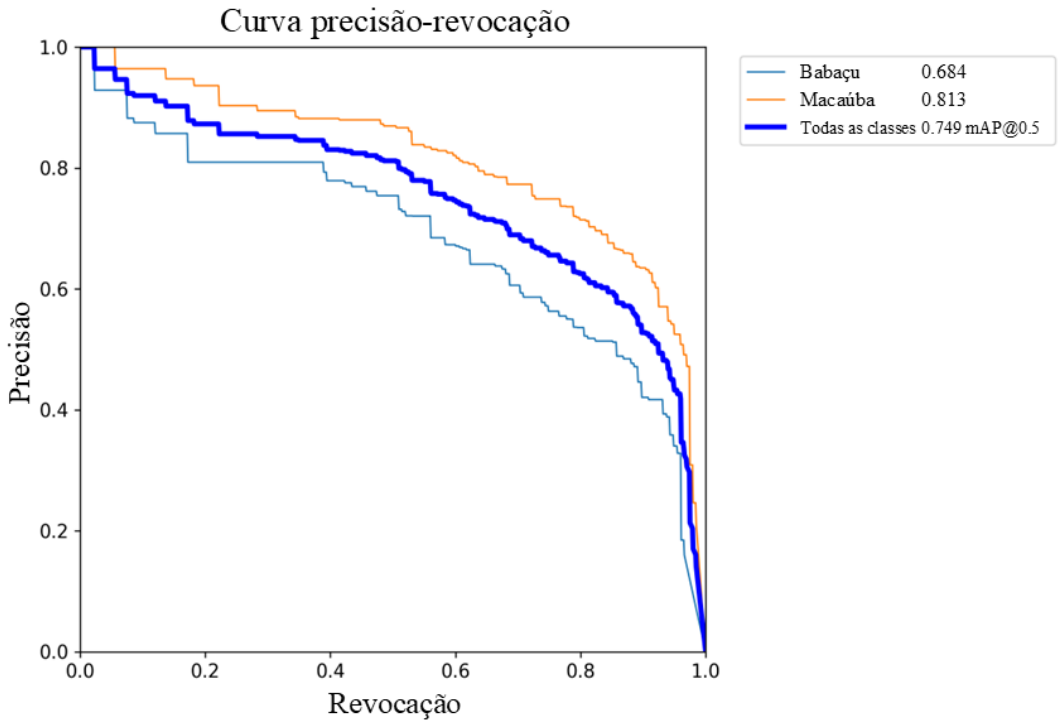

Figura B.15: Curva precisão-revocação relativa à versão Y9-6 da YOLOv9.

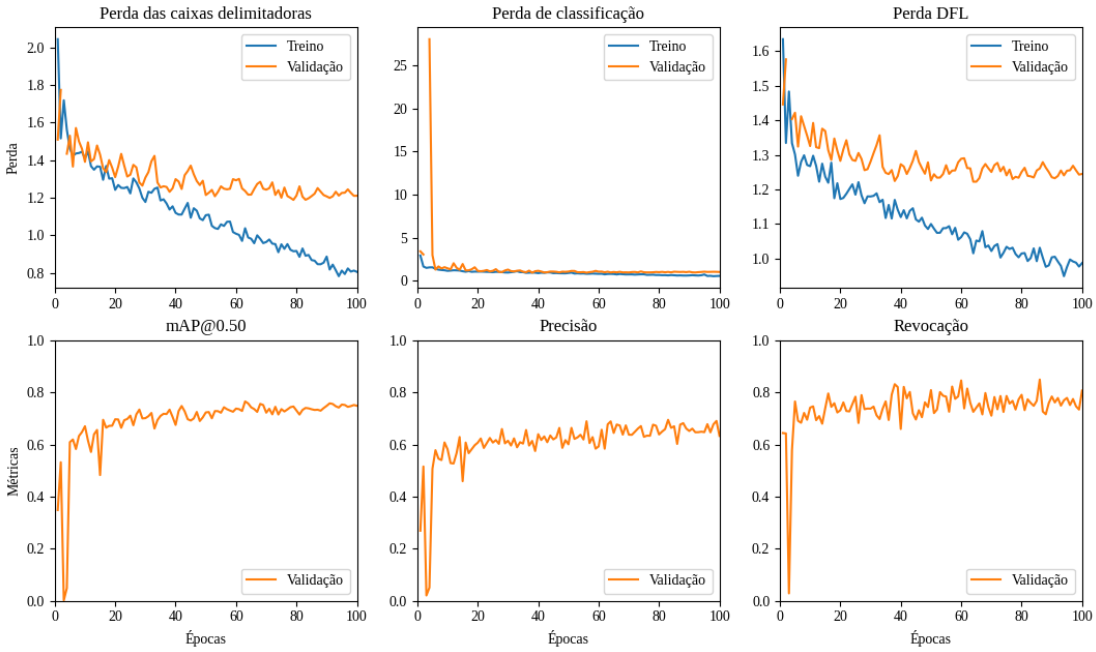

Figura B.16: Gráficos das perdas das caixas delimitadoras, de classificação e DFL; mAP@ 0.50, precisão e revocação em função das épocas relativo ao treinamento da versão Y9-7 utilizando a YOLOv9.

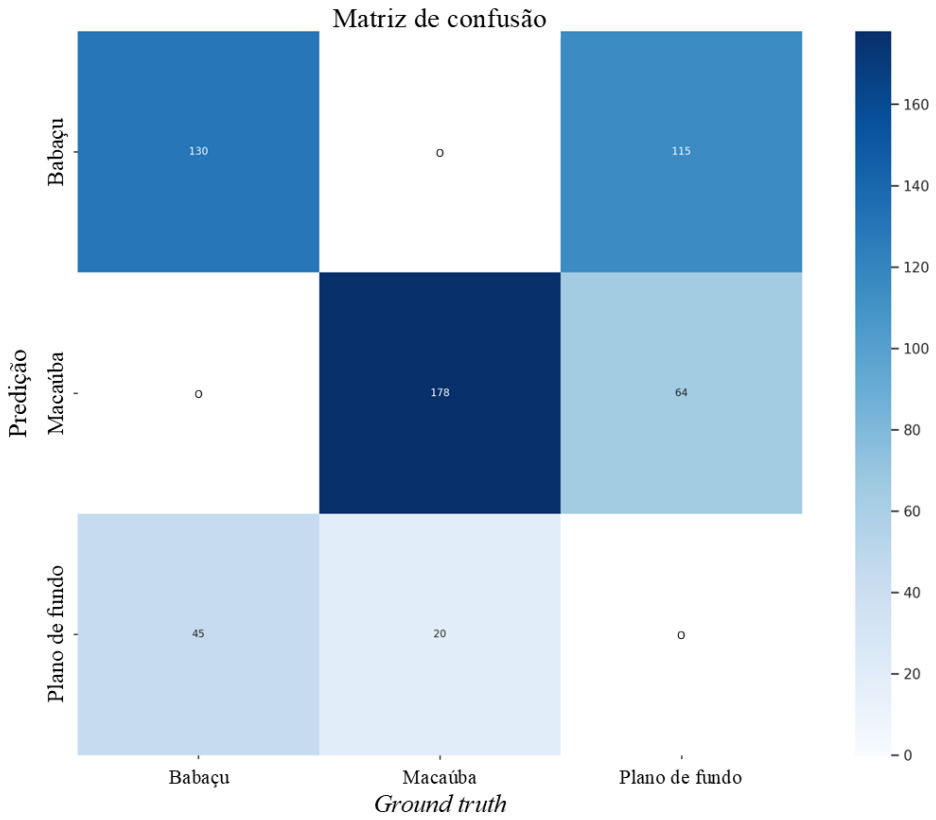

Figura B.17: Matriz de confusão relativa à versão Y9-7 da YOLOv9.

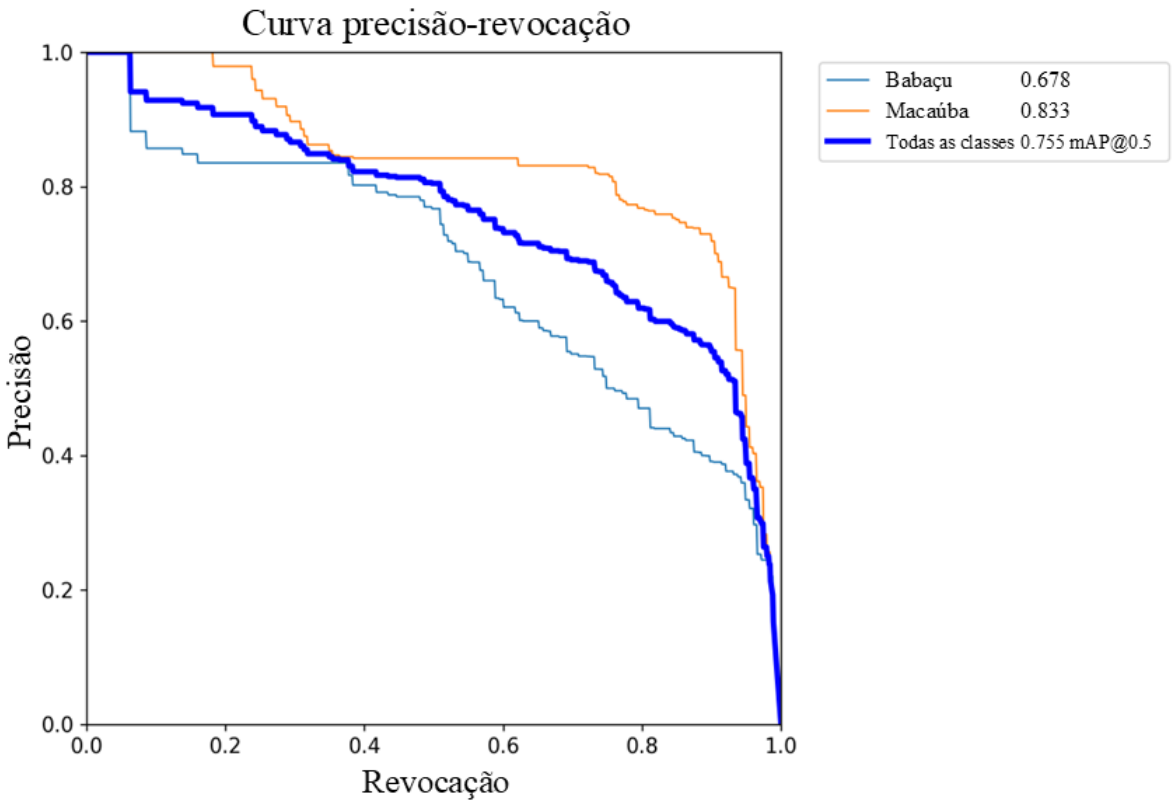

Figura B.18: Curva precisão-revocação relativa à versão Y9-7 da YOLOv9.

B.2 Gráficos relativos ao teste de validação cruzada *k-fold*

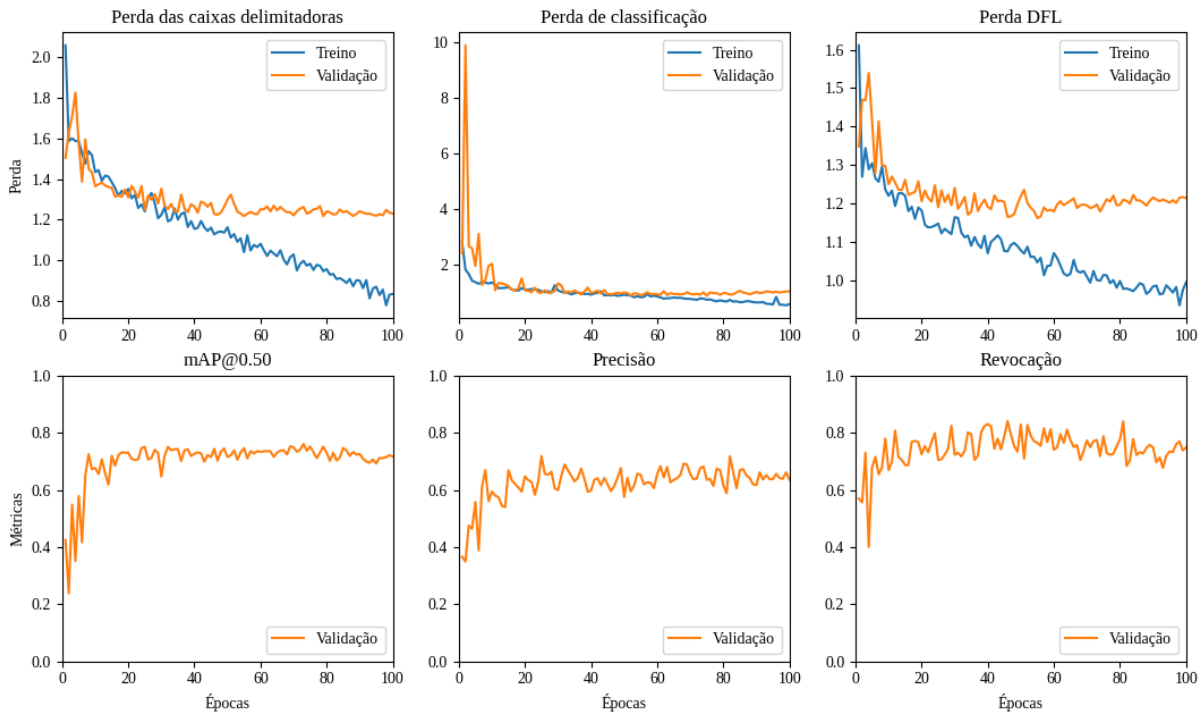

Figura B.19: Gráficos das perdas das caixas delimitadoras, de classificação e DFL; mAP@0.50, precisão e revocação em função das épocas relativo ao treinamento do *fold* 1 da validação cruzada *k-fold* da YOLOv9.

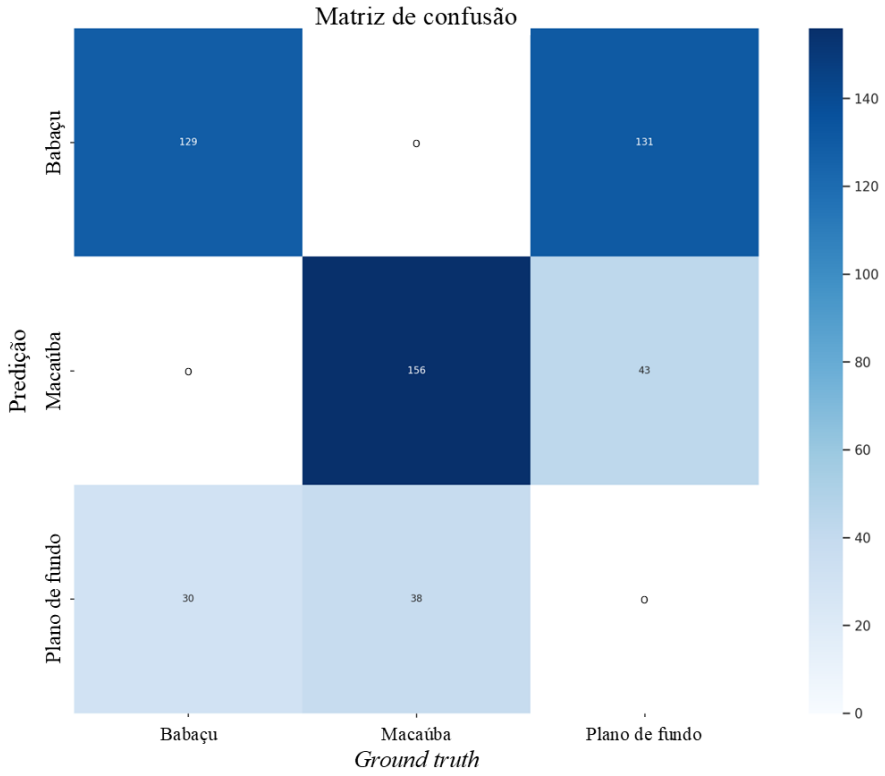

Figura B.20: Matriz de confusão relativa ao treinamento do *fold* 1 da validação cruzada *k-fold* da YOLOv9.

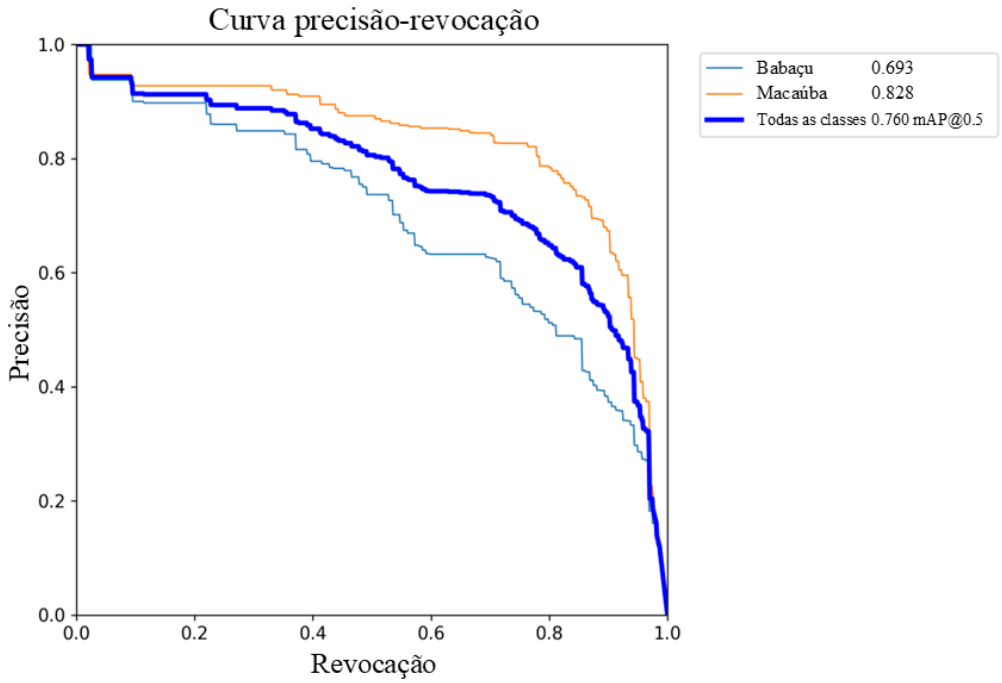

Figura B.21: Curva precisão-revocação relativa ao treinamento do *fold* 1 da validação cruzada *k-fold* da YOLOv9.

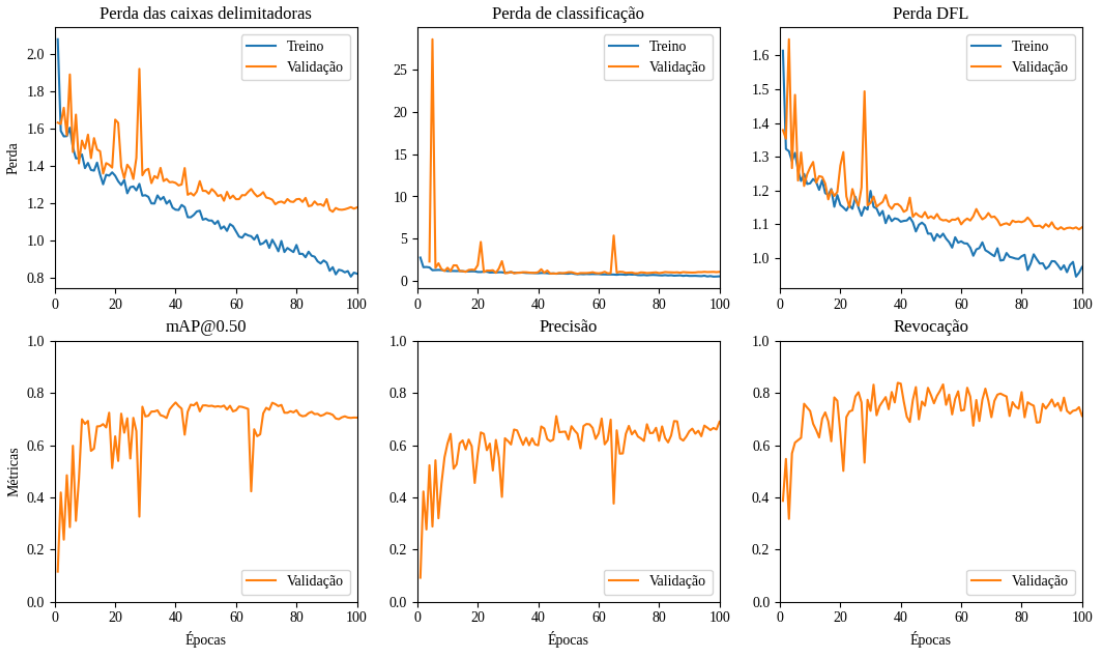

Figura B.22: Gráficos das perdas das caixas delimitadoras, de classificação e DFL; mAP@0.50, precisão e revocação em função das épocas relativo ao treinamento do *fold* 2 da validação cruzada *k-fold* da YOLOv9.

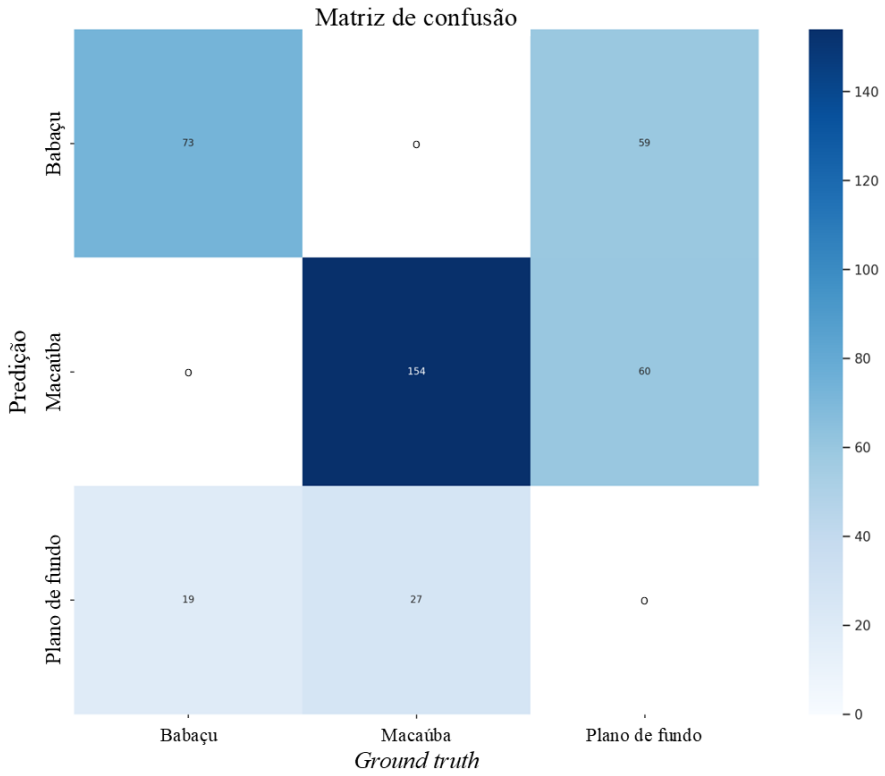

Figura B.23: Matriz de confusão relativa ao treinamento do *fold* 2 da validação cruzada *k-fold* da YOLOv9.

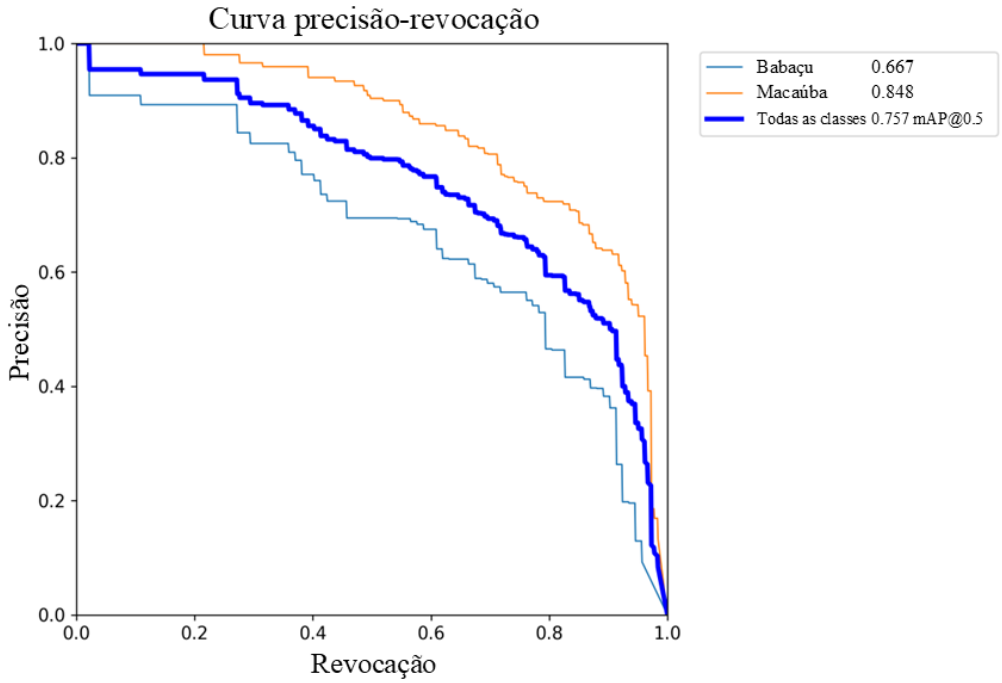

Figura B.24: Curva precisão-revocação relativa ao treinamento do *fold* 2 da validação cruzada *k-fold* da YOLOv9.

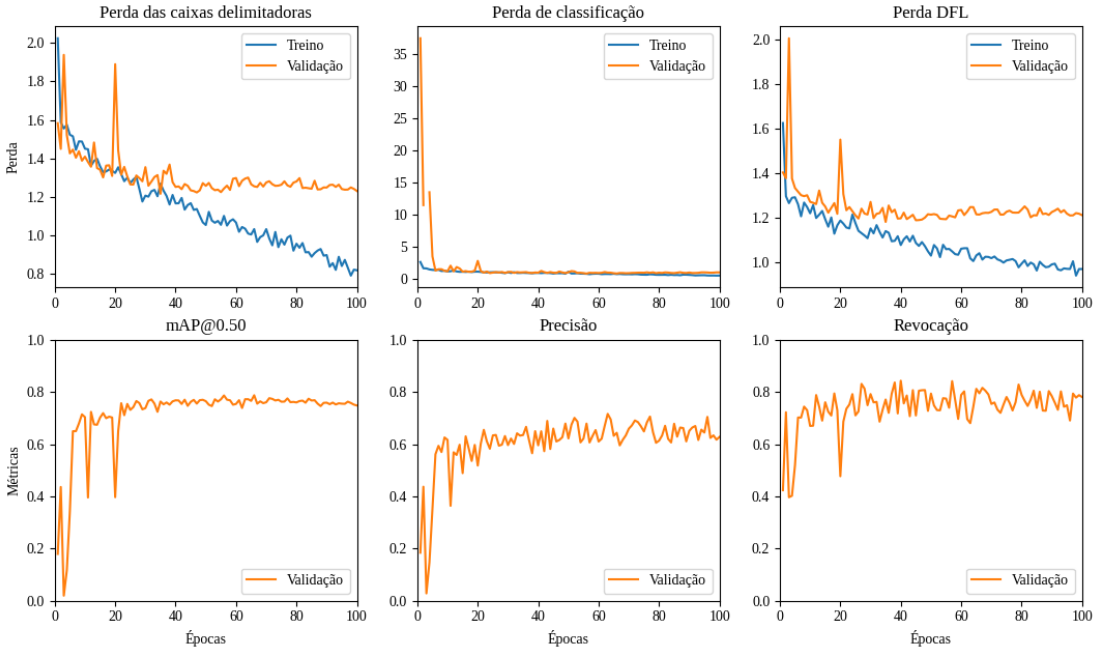

Figura B.25: Gráficos das perdas das caixas delimitadoras, de classificação e DFL; mAP@ 0.50, precisão e revocação em função das épocas relativo ao treinamento do *fold* 3 da validação cruzada *k-fold* da YOLOv9.

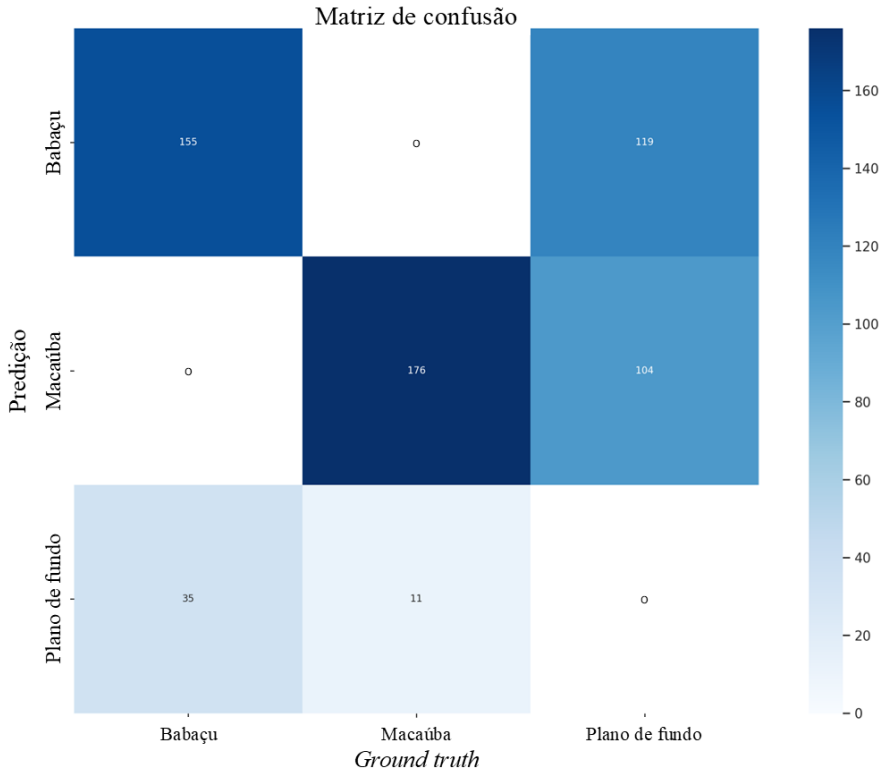

Figura B.26: Matriz de confusão relativa ao treinamento do *fold* 3 da validação cruzada *k-fold* da YOLOv9.

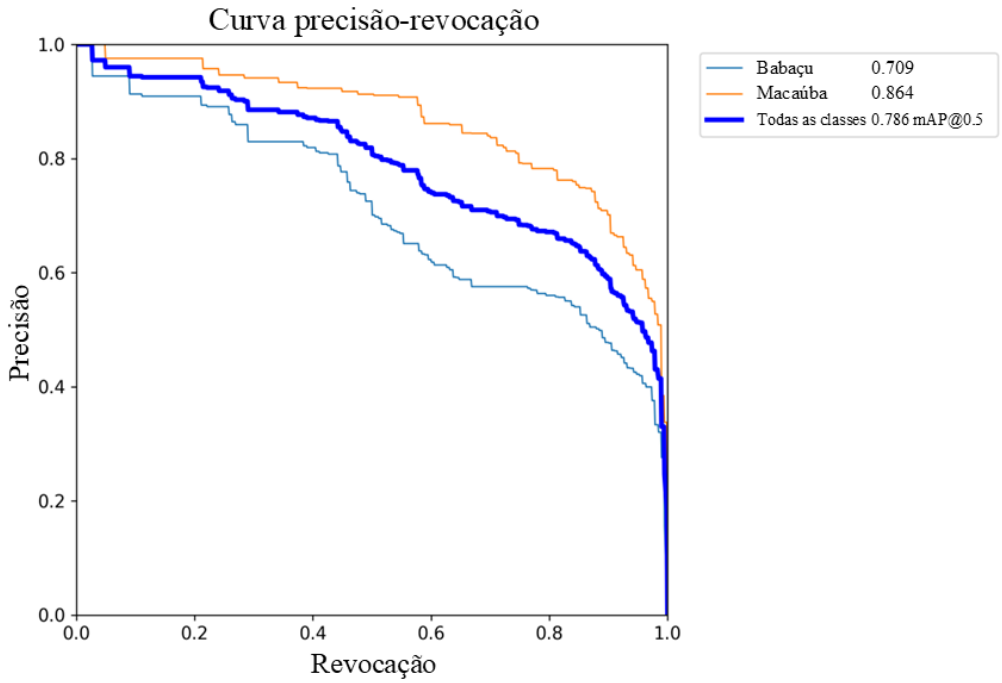

Figura B.27: Curva precisão-revocação relativa ao treinamento do *fold* 3 da validação cruzada *k-fold* da YOLOv9.

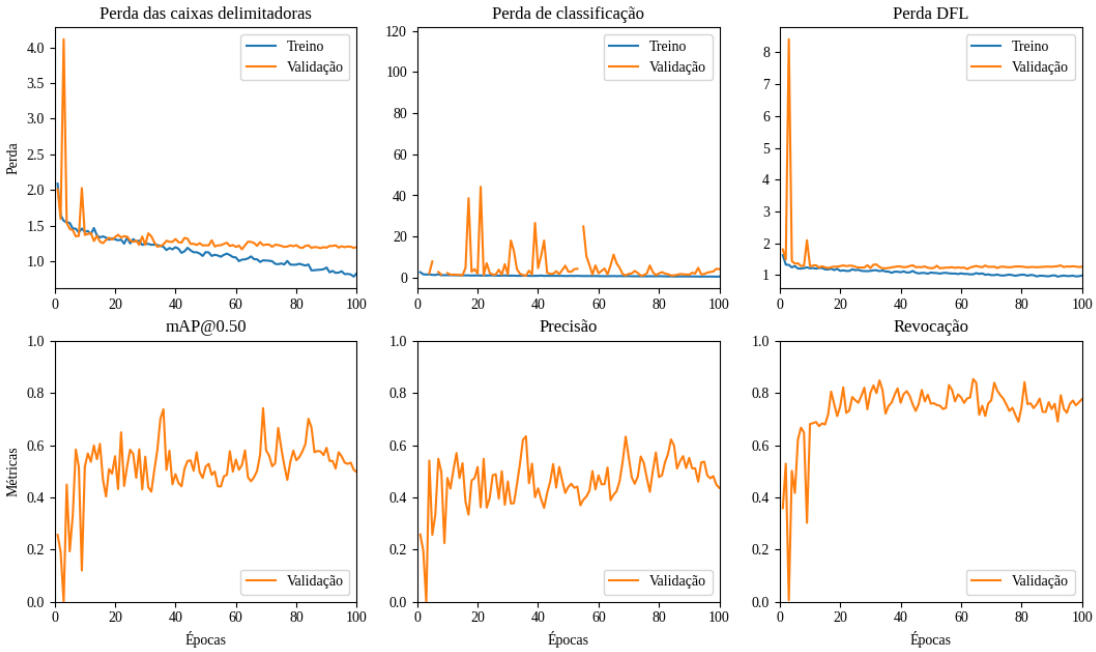

Figura B.28: Gráficos das perdas das caixas delimitadoras, de classificação e DFL; mAP@0.50, precisão e revocação em função das épocas relativo ao treinamento do *fold* 4 da validação cruzada *k-fold* da YOLOv9.

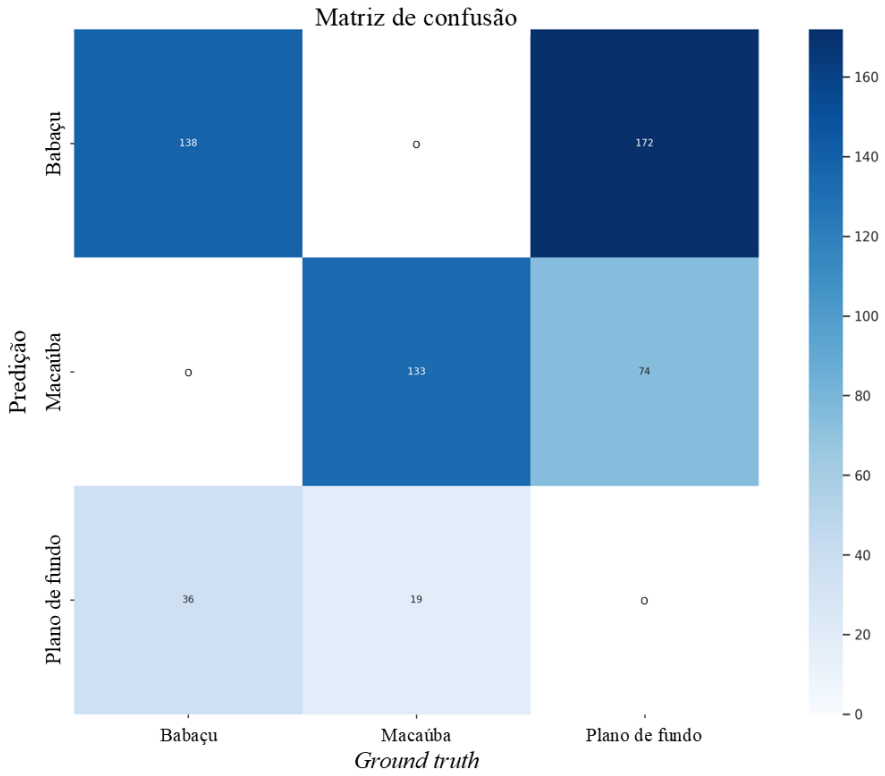

Figura B.29: Matriz de confusão relativa ao treinamento do *fold* 4 da validação cruzada *k-fold* da YOLOv9.

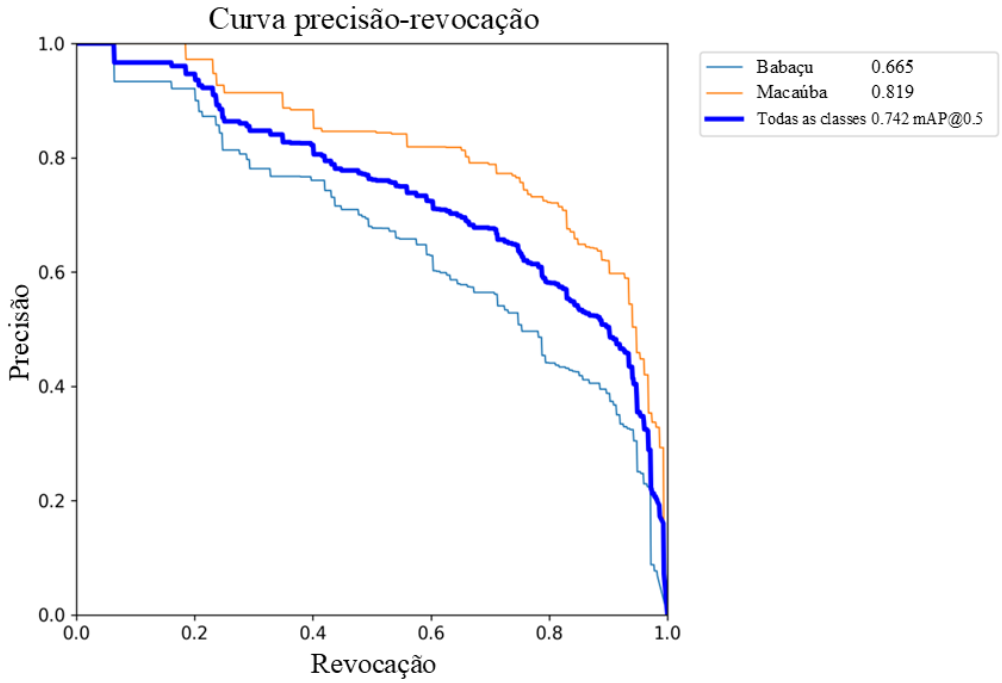

Figura B.30: Curva precisão-revocação relativa ao treinamento do *fold* 4 da validação cruzada *k-fold* da YOLOv9.

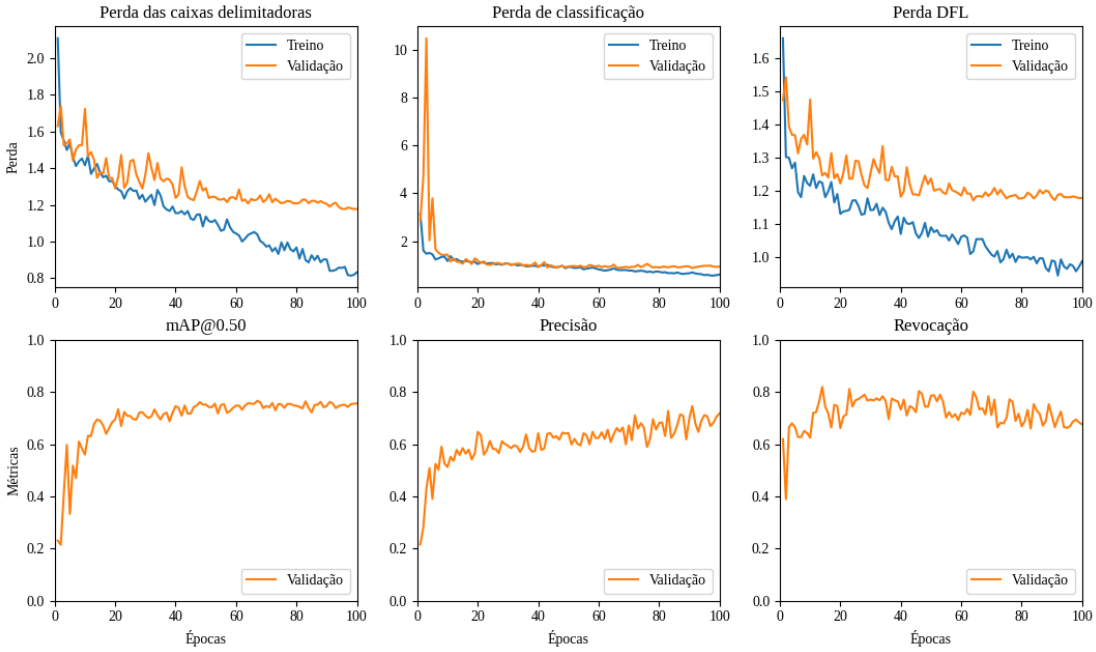

Figura B.31: Gráficos das perdas das caixas delimitadoras, de classificação e DFL; mAP@ 0.50, precisão e revocação em função das épocas relativo ao treinamento do *fold* 5 da validação cruzada *k-fold* da YOLOv9.

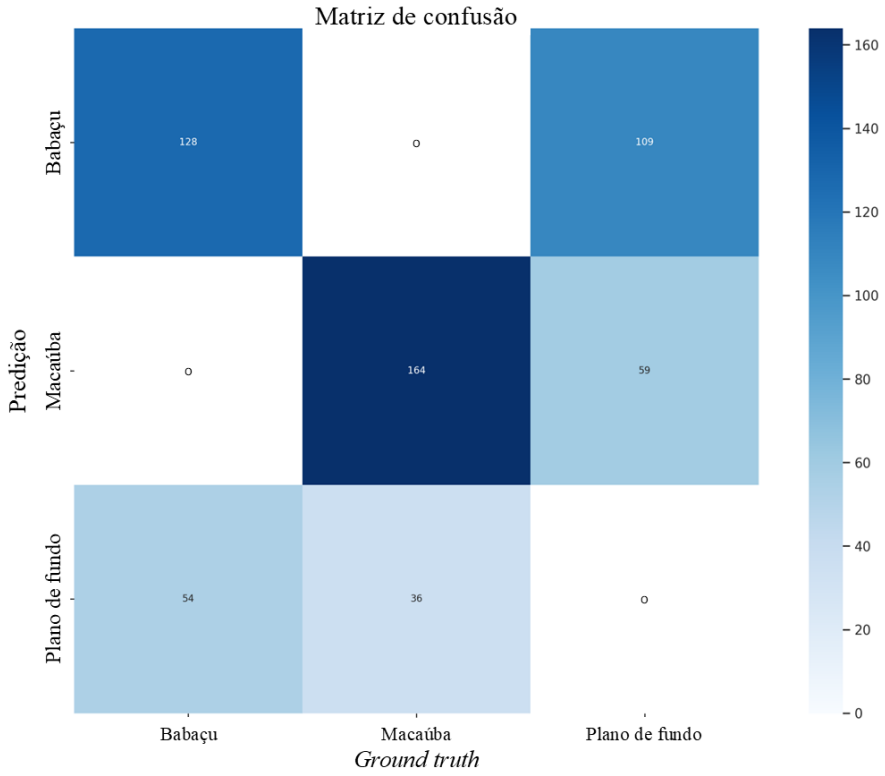

Figura B.32: Matriz de confusão relativa ao treinamento do *fold* 5 da validação cruzada *k-fold* da YOLOv9.

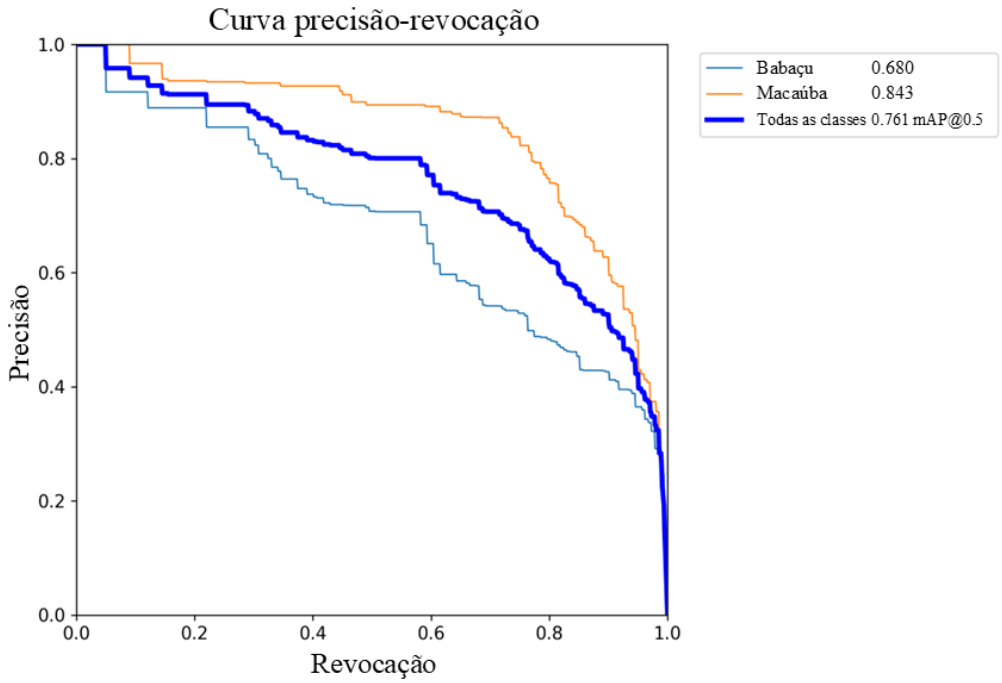

Figura B.33: Curva precisão-revocação relativa ao treinamento do *fold* 5 da validação cruzada *k-fold* da YOLOv9.

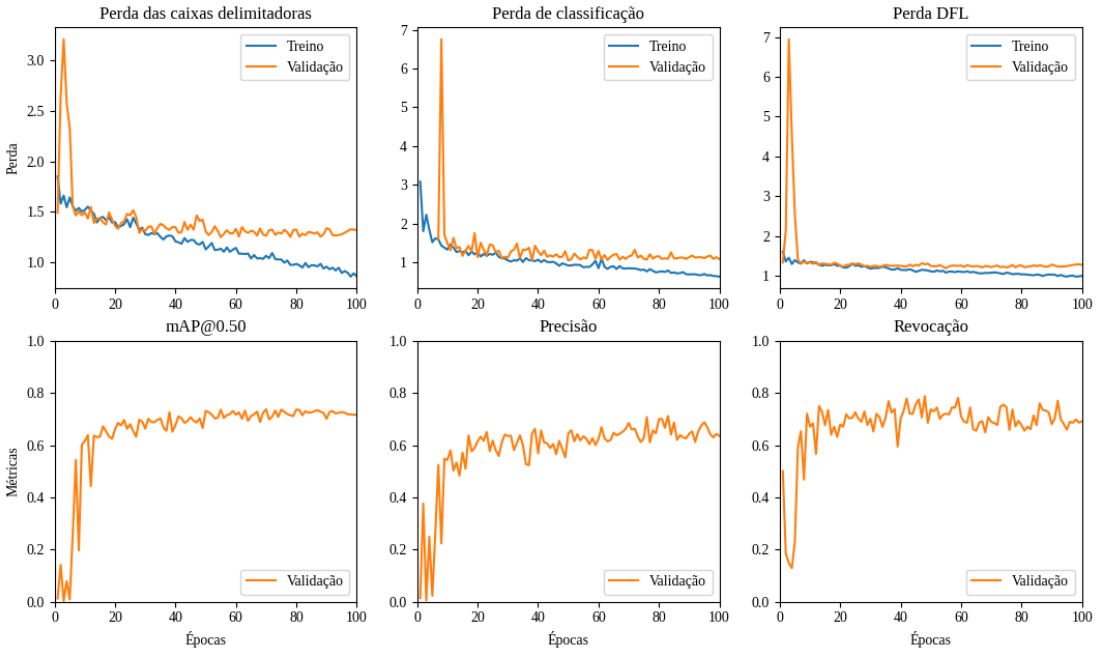

Figura B.34: Gráficos das perdas das caixas delimitadoras, de classificação e DFL; mAP@0.50, precisão e revocação em função das épocas relativo ao treinamento do *fold* 7 da validação cruzada *k-fold* da YOLOv9.

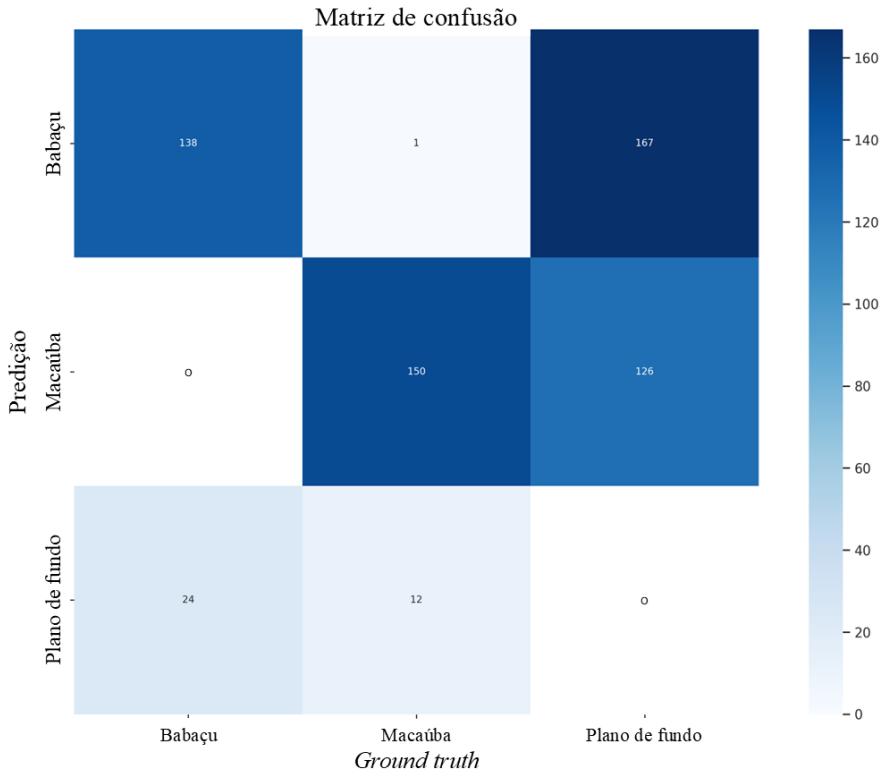

Figura B.35: Matriz de confusão relativa ao treinamento do *fold* 7 da validação cruzada *k-fold* da YOLOv9.

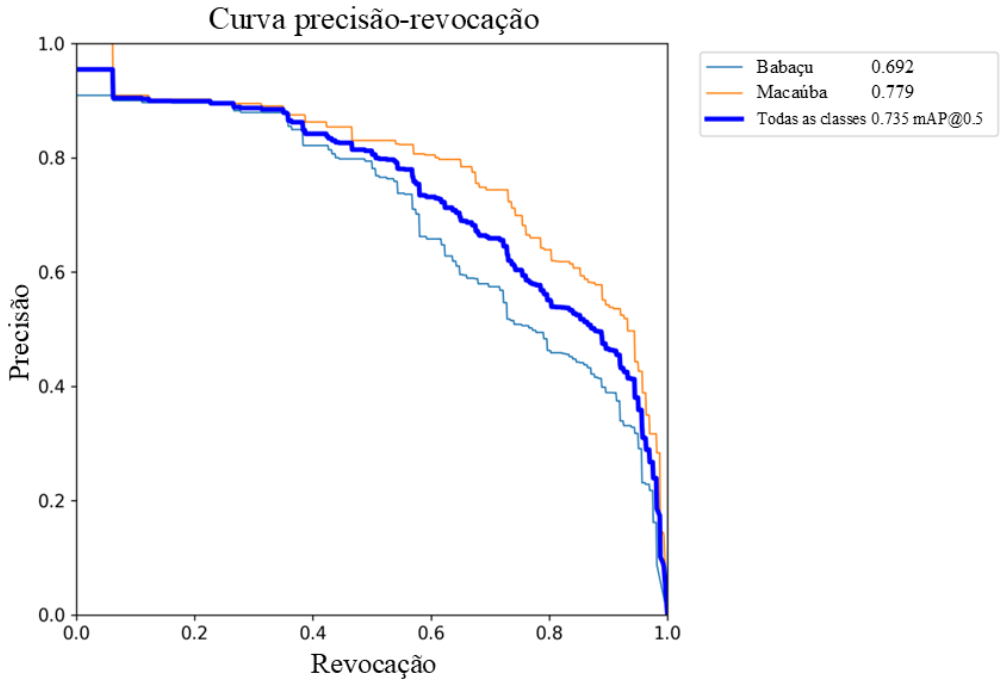

Figura B.36: Curva precisão-revocação relativa ao treinamento do *fold* 7 da validação cruzada *k-fold* da YOLOv9.

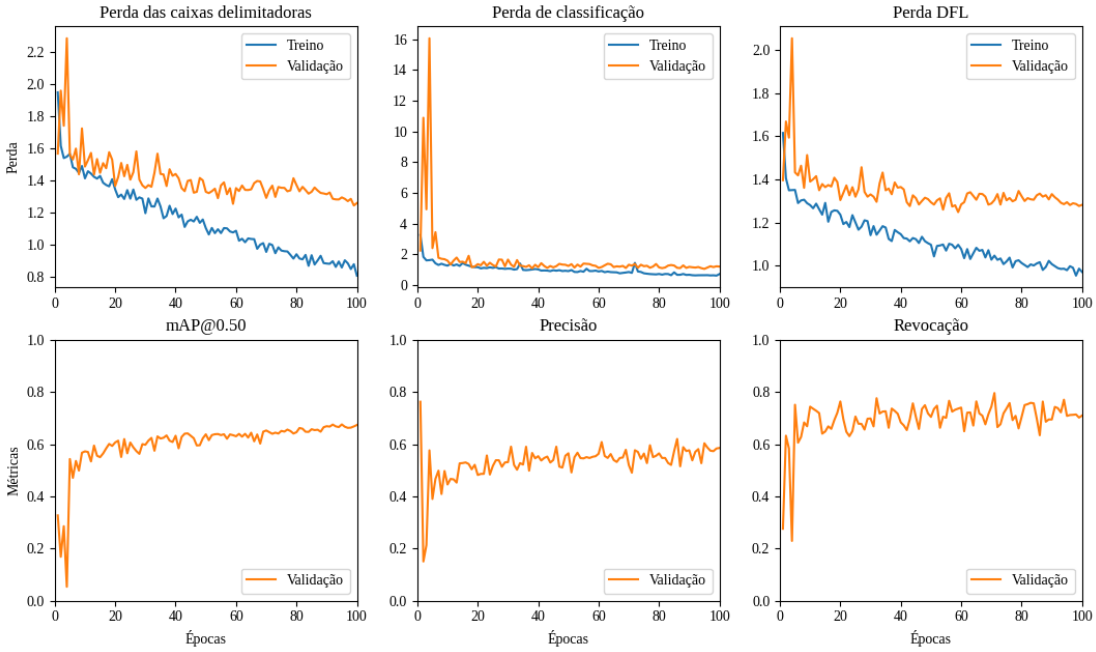

Figura B.37: Gráficos das perdas das caixas delimitadoras, de classificação e DFL; mAP@ 0.50, precisão e revocação em função das épocas relativo ao treinamento do *fold* 8 da validação cruzada *k-fold* da YOLOv9.

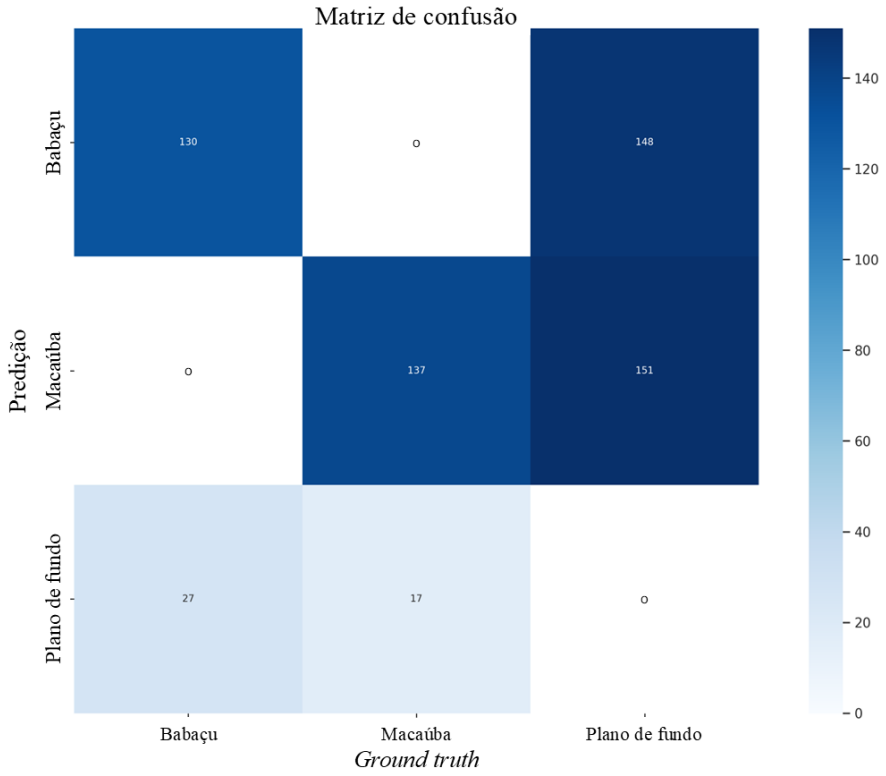

Figura B.38: Matriz de confusão relativa ao treinamento do *fold* 8 da validação cruzada *k-fold* da YOLOv9.

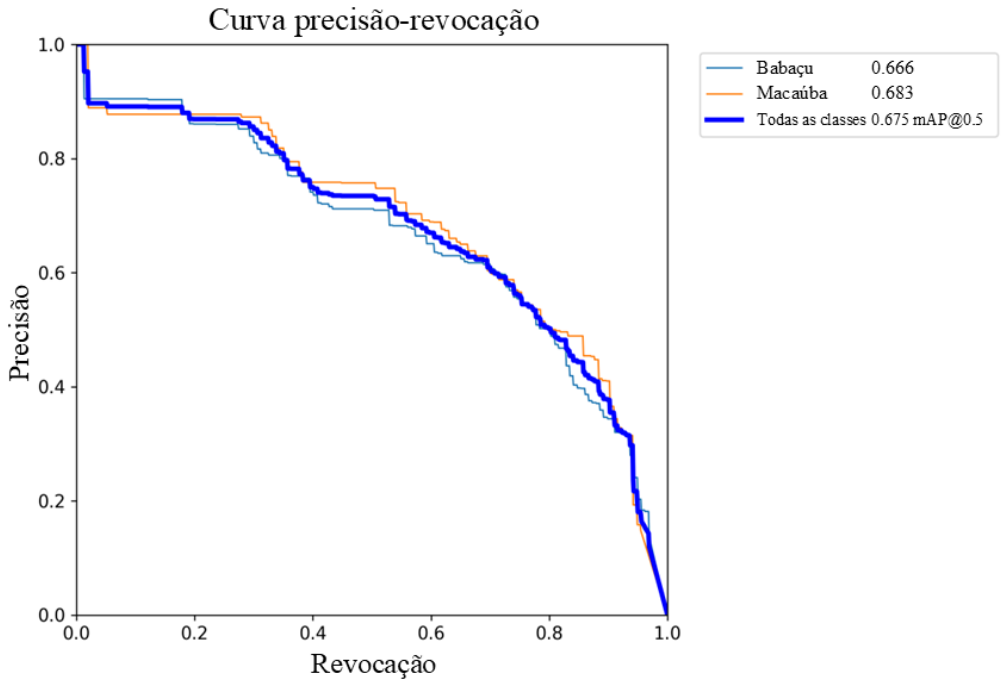

Figura B.39: Curva precisão-revocação relativa ao treinamento do *fold* 8 da validação cruzada *k-fold* da YOLOv9.

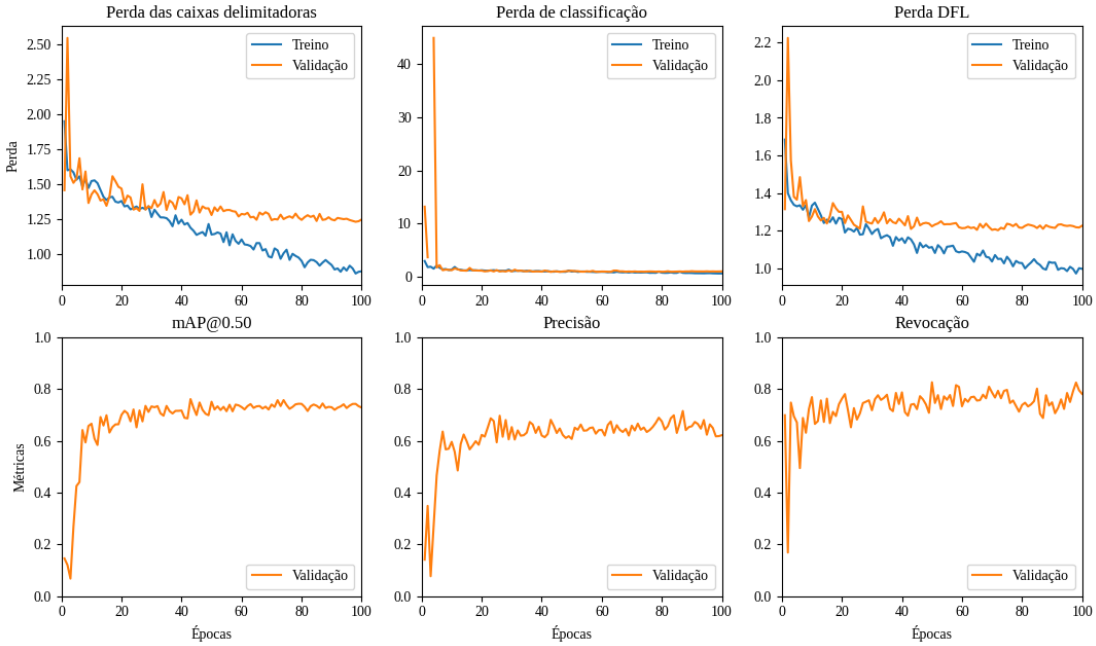

Figura B.40: Gráficos das perdas das caixas delimitadoras, de classificação e DFL; mAP@0.50, precisão e revocação em função das épocas relativo ao treinamento do *fold* 9 da validação cruzada *k-fold* da YOLOv9.

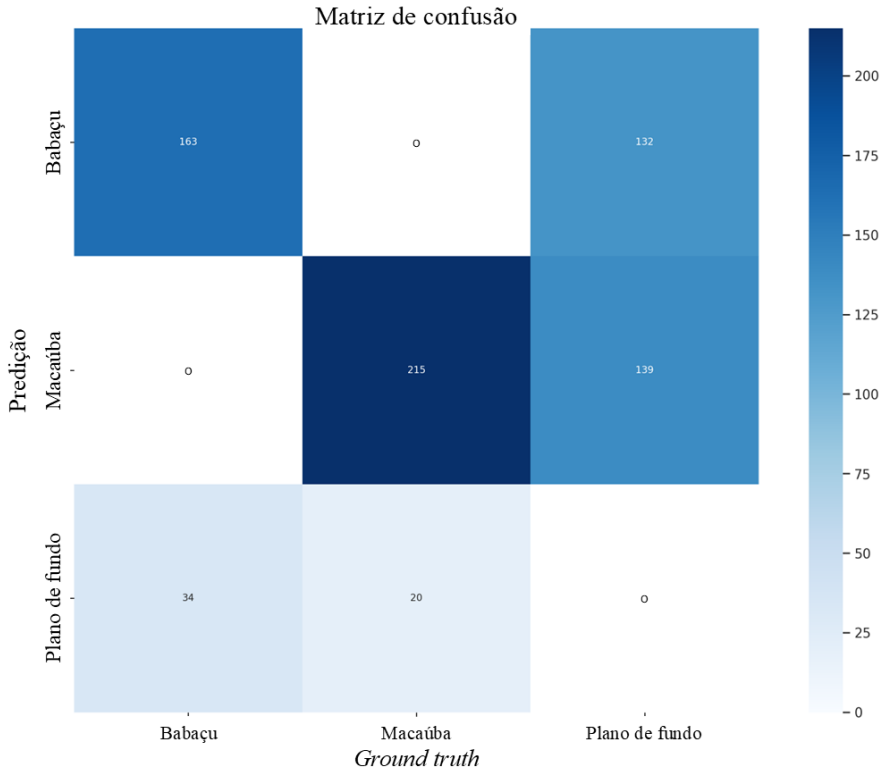

Figura B.41: Matriz de confusão relativa ao treinamento do *fold* 9 da validação cruzada *k-fold* da YOLOv9.

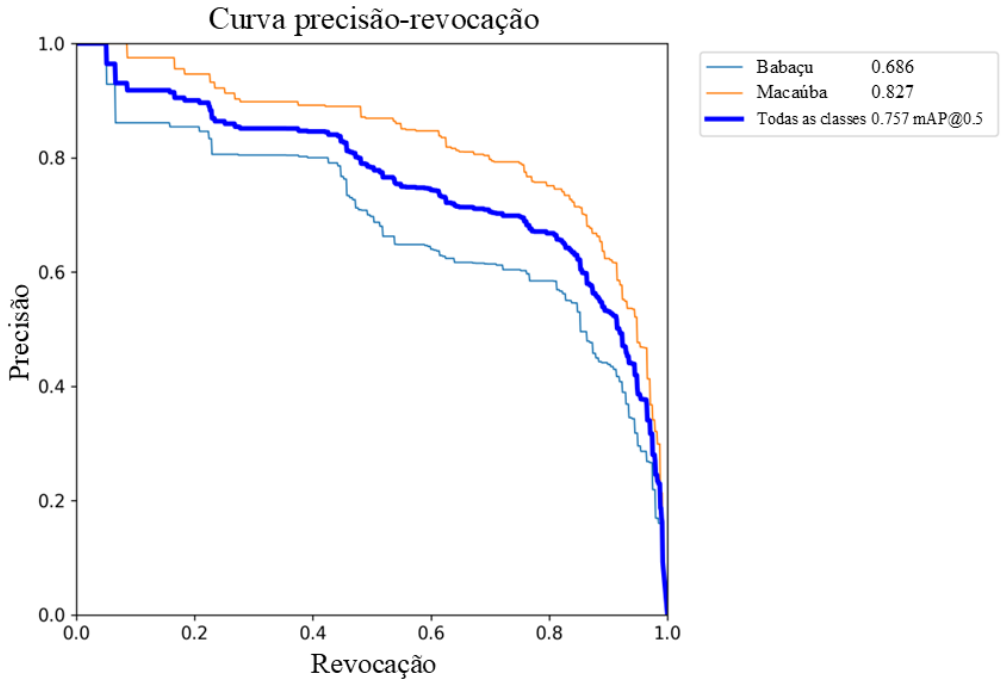

Figura B.42: Curva precisão-revocação relativa ao treinamento do *fold* 9 da validação cruzada *k-fold* da YOLOv9.

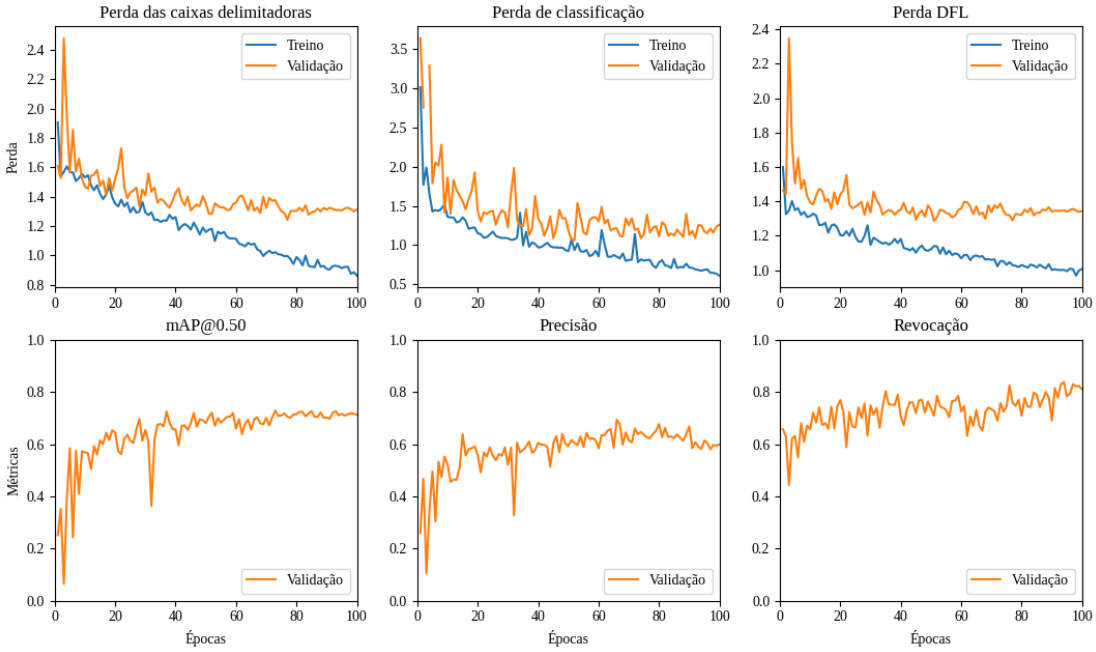

Figura B.43: Gráficos das perdas das caixas delimitadoras, de classificação e DFL; mAP@ 0.50, precisão e revocação em função das épocas relativo ao treinamento do *fold* 10 da validação cruzada *k-fold* da YOLOv9.

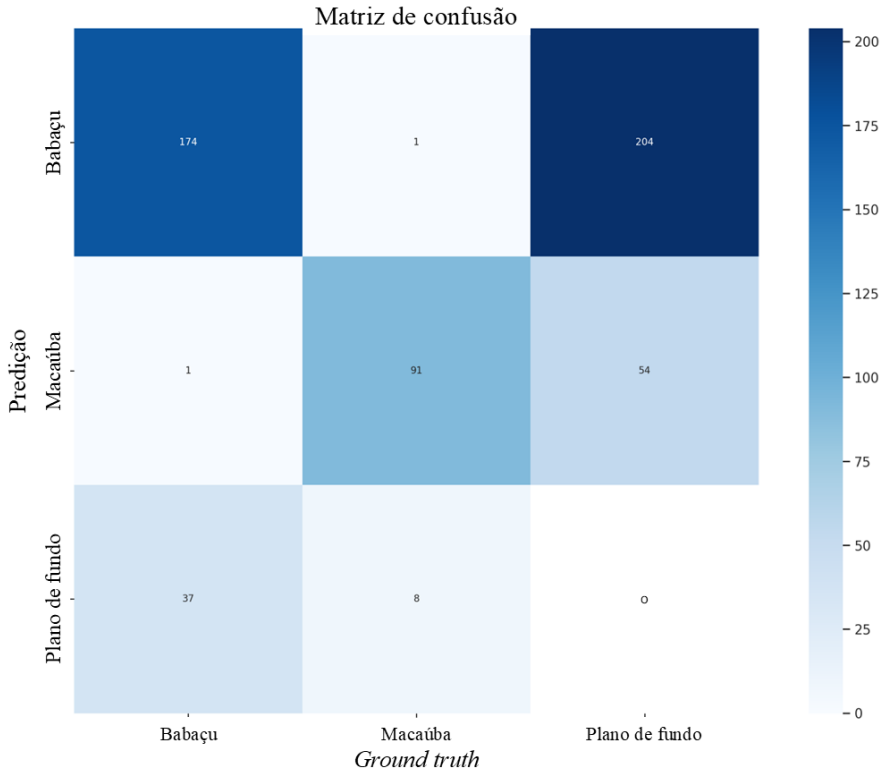

Figura B.44: Matriz de confusão relativa ao treinamento do *fold* 10 da validação cruzada *k-fold* da YOLOv9.

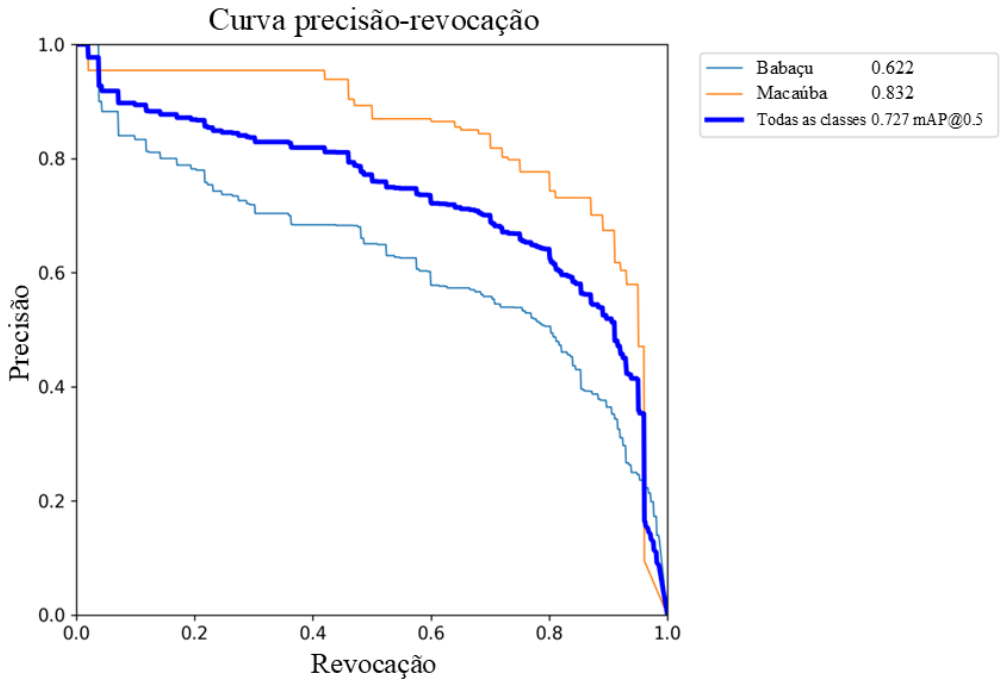

Figura B.45: Curva precisão-revocação relativa ao treinamento do *fold* 10 da validação cruzada *k-fold* da YOLOv9.

Apêndice C

Lista das imagens utilizadas e suas aplicações

Tabela C.1: Relação das imagens utilizadas para o treinamento dos modelos de *auto labeling* (AL v. 1 a 5), produção (utilizando pesos pré-treinados do conjunto de dados MS COCO) e produção utilizando pesos pré-treinados do sistema de *auto labeling*.

Imagens	AL v. 1	AL v. 2	AL v. 3	AL v. 4	AL v. 5	Produção	Produção AL
A01_Split20000-0-0_patch_168.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_011.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_012.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_024.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_025.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_036.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_037.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_038.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_049.jpg	Teste	Teste	Teste	Teste	Teste	Treino	Teste
A01_Split20000-0-1_patch_050.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_051.jpg	Teste	Teste	Teste	Teste	Teste	Treino	Teste
A01_Split20000-0-1_patch_062.jpg	N/A	Treino	Treino	Treino	Treino	Teste	Treino
A01_Split20000-0-1_patch_063.jpg	N/A	Treino	Treino	Treino	Treino	Teste	Treino
A01_Split20000-0-1_patch_064.jpg	Treino	Treino	Teste	Teste	Teste	Treino	Teste
A01_Split20000-0-1_patch_075.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_076.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino
A01_Split20000-0-1_patch_077.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_086.jpg	Teste	Teste	Teste	Teste	Teste	Treino	Teste
A01_Split20000-0-1_patch_087.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_088.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_089.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_090.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino
A01_Split20000-0-1_patch_099.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_100.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_101.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_102.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_103.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_111.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_112.jpg	Treino	Treino	Treino	Teste	Teste	Treino	Teste
A01_Split20000-0-1_patch_113.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino
A01_Split20000-0-1_patch_114.jpg	Treino	Treino	Treino	Teste	Teste	Treino	Teste

Imagens	AL v. 1	AL v. 2	AL v. 3	AL v. 4	AL v. 5	Produção	Produção AL
A01_Split20000-0-1_patch_115.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_116.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_125.jpg	Teste	Teste	Teste	Teste	Teste	Treino	Teste
A01_Split20000-0-1_patch_126.jpg	Treino	Treino	Treino	Teste	Teste	Treino	Teste
A01_Split20000-0-1_patch_127.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino
A01_Split20000-0-1_patch_128.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino
A01_Split20000-0-1_patch_129.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_138.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_139.jpg	Teste	Teste	Teste	Teste	Teste	Treino	Teste
A01_Split20000-0-1_patch_140.jpg	Treino	Teste	Teste	Teste	Teste	Teste	Teste
A01_Split20000-0-1_patch_141.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_142.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_150.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino
A01_Split20000-0-1_patch_151.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino
A01_Split20000-0-1_patch_152.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_153.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_154.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_155.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_163.jpg	N/A	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_164.jpg	N/A	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_165.jpg	N/A	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_166.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_167.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_168.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_006.jpg	N/A	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_007.jpg	N/A	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_008.jpg	N/A	Treino	Treino	Treino	Treino	Treino	Teste
A01_Split20000-0-2_patch_009.jpg	N/A	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_010.jpg	N/A	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_011.jpg	N/A	Treino	Treino	Treino	Treino	Teste	Treino
A01_Split20000-0-2_patch_012.jpg	N/A	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_019.jpg	N/A	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_020.jpg	N/A	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_021.jpg	N/A	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_022.jpg	N/A	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_023.jpg	N/A	Treino	Treino	Treino	Teste	Treino	Teste
A01_Split20000-0-2_patch_024.jpg	N/A	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_025.jpg	N/A	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_030.jpg	N/A	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_031.jpg	N/A	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_032.jpg	N/A	Teste	Teste	Teste	Teste	Treino	Teste
A01_Split20000-0-2_patch_033.jpg	N/A	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_034.jpg	N/A	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_035.jpg	N/A	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_036.jpg	N/A	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_037.jpg	N/A	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_038.jpg	N/A	Treino	Teste	Teste	Teste	Treino	Teste
A01_Split20000-0-2_patch_043.jpg	N/A	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_044.jpg	N/A	Teste	Teste	Teste	Teste	Treino	Teste
A01_Split20000-0-2_patch_045.jpg	N/A	Treino	Treino	Treino	Treino	Teste	Treino
A01_Split20000-0-2_patch_046.jpg	N/A	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_047.jpg	N/A	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_048.jpg	N/A	Teste	Teste	Teste	Teste	Treino	Teste
A01_Split20000-0-2_patch_049.jpg	N/A	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_050.jpg	N/A	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_051.jpg	N/A	Treino	Treino	Treino	Treino	Treino	Treino

Imagens	AL v. 1	AL v. 2	AL v. 3	AL v. 4	AL v. 5	Produção	Produção AL
A01_Split20000-0-2_patch_055.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
A01_Split20000-0-2_patch_056.jpg	N/A	Treino	Treino	Treino	Teste	Teste	Teste
A01_Split20000-0-2_patch_057.jpg	N/A	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_058.jpg	N/A	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_059.jpg	N/A	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_060.jpg	N/A	Treino	Treino	Treino	Treino	Teste	Treino
A01_Split20000-0-2_patch_061.jpg	N/A	Treino	Treino	Treino	Teste	Teste	Teste
A01_Split20000-0-2_patch_062.jpg	N/A	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_063.jpg	N/A	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_064.jpg	N/A	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_068.jpg	N/A	Treino	Teste	Teste	Teste	Treino	Teste
A01_Split20000-0-2_patch_069.jpg	N/A	Treino	Teste	Teste	Teste	Treino	Teste
A01_Split20000-0-2_patch_070.jpg	N/A	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_071.jpg	N/A	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_072.jpg	N/A	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_073.jpg	N/A	Treino	Treino	Teste	Teste	Treino	Teste
A01_Split20000-0-2_patch_074.jpg	N/A	Treino	Treino	Treino	Treino	Teste	Treino
A01_Split20000-0-2_patch_075.jpg	N/A	Teste	Teste	Teste	Teste	Treino	Teste
A01_Split20000-0-2_patch_076.jpg	N/A	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_077.jpg	N/A	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_081.jpg	N/A	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_082.jpg	N/A	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_083.jpg	N/A	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_084.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Teste
A01_Split20000-0-2_patch_085.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
A01_Split20000-0-2_patch_086.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
A01_Split20000-0-2_patch_088.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_089.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
A01_Split20000-0-2_patch_090.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_094.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_098.jpg	N/A	N/A	N/A	Treino	Treino	Teste	Treino
A01_Split20000-0-2_patch_100.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_102.jpg	N/A	N/A	N/A	N/A	N/A	Teste	Treino
A01_Split20000-0-2_patch_106.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
A01_Split20000-0-2_patch_126.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
A01_Split20000-0-2_patch_128.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
A01_Split20000-0-2_patch_137.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
A01_Split20000-0-2_patch_138.jpg	N/A	N/A	Treino	Treino	Treino	Teste	Treino
A01_Split20000-0-2_patch_145.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_146.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_147.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_153.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_158.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_159.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
A01_Split20000-0-2_patch_162.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_163.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_164.jpg	N/A	N/A	N/A	N/A	N/A	Teste	Treino
A01_Split20000-0-2_patch_166.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
A01_Split20000-0-3_patch_025.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
A01_Split20000-0-3_patch_038.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
A01_Split20000-0-3_patch_051.jpg	N/A	N/A	N/A	Treino	Treino	Teste	Treino
A01_Split20000-1-0_patch_017.jpg	N/A	N/A	Treino	Treino	Treino	Teste	Treino
A01_Split20000-1-0_patch_031.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_043.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
A01_Split20000-1-0_patch_045.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
A01_Split20000-1-0_patch_046.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino

Imagens	AL v. 1	AL v. 2	AL v. 3	AL v. 4	AL v. 5	Produção	Produção AL
A01_Split20000-1-0_patch_057.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
A01_Split20000-1-0_patch_058.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
A01_Split20000-1-0_patch_060.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_067.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_073.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_080.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
A01_Split20000-1-0_patch_084.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_086.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
A01_Split20000-1-0_patch_094.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
A01_Split20000-1-0_patch_095.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
A01_Split20000-1-0_patch_096.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
A01_Split20000-1-0_patch_097.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
A01_Split20000-1-0_patch_102.jpg	N/A	N/A	Teste	Teste	Teste	Treino	Teste
A01_Split20000-1-0_patch_112.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_113.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
A01_Split20000-1-0_patch_118.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
A01_Split20000-1-0_patch_121.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
A01_Split20000-1-0_patch_126.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
A01_Split20000-1-0_patch_127.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
A01_Split20000-1-0_patch_130.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
A01_Split20000-1-0_patch_131.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
A01_Split20000-1-0_patch_132.jpg	N/A	N/A	N/A	N/A	Treino	Teste	Treino
A01_Split20000-1-0_patch_142.jpg	N/A	N/A	Treino	Treino	Treino	Teste	Treino
A01_Split20000-1-0_patch_144.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_153.jpg	N/A	N/A	N/A	N/A	Treino	Teste	Treino
A01_Split20000-1-0_patch_164.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
A01_Split20000-1-0_patch_165.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Teste
A01_Split20000-1-0_patch_168.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
A01_Split20000-1-1_patch_006.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
A01_Split20000-1-1_patch_011.jpg	N/A	N/A	N/A	N/A	N/A	Teste	Treino
A01_Split20000-1-1_patch_012.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_013.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_022.jpg	N/A	N/A	N/A	N/A	Teste	Treino	Teste
A01_Split20000-1-1_patch_024.jpg	N/A	N/A	Teste	Teste	Teste	Treino	Teste
A01_Split20000-1-1_patch_031.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_035.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
A01_Split20000-1-1_patch_037.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_044.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
A01_Split20000-1-1_patch_045.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
A01_Split20000-1-1_patch_046.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
A01_Split20000-1-1_patch_048.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
A01_Split20000-1-1_patch_055.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_058.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_060.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
A01_Split20000-1-1_patch_061.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_063.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_072.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_073.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
A01_Split20000-1-1_patch_082.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_085.jpg	N/A	N/A	N/A	N/A	N/A	Teste	Treino
A01_Split20000-1-1_patch_100.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
A01_Split20000-1-1_patch_104.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_105.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
A01_Split20000-1-1_patch_108.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
A01_Split20000-1-1_patch_118.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_122.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino

Imagens	AL v. 1	AL v. 2	AL v. 3	AL v. 4	AL v. 5	Produção	Produção AL
A01_Split20000-1-1_patch_125.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
A01_Split20000-1-1_patch_127.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_133.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_136.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
A01_Split20000-1-1_patch_142.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
A01_Split20000-1-1_patch_147.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
A01_Split20000-1-1_patch_156.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
A01_Split20000-1-1_patch_157.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-2_patch_003.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
A01_Split20000-1-2_patch_008.jpg	N/A	N/A	Treino	Treino	Treino	N/A	N/A
A01_Split20000-1-2_patch_009.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
A01_Split20000-1-2_patch_016.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
A01_Split20000-1-2_patch_021.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-2_patch_030.jpg	N/A	N/A	N/A	Treino	Treino	Teste	Treino
A01_Split20000-1-2_patch_046.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
A01_Split20000-1-2_patch_047.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
A01_Split20000-1-2_patch_053.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
A01_Split20000-1-2_patch_057.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
A01_Split20000-1-2_patch_058.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
A01_Split20000-1-2_patch_059.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
A01_Split20000-1-2_patch_074.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
A01_Split20000-1-2_patch_084.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
A01_Split20000-1-2_patch_085.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-2_patch_086.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
A01_Split20000-1-2_patch_092.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
A01_Split20000-1-2_patch_099.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-2_patch_106.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
A01_Split20000-1-2_patch_110.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-2_patch_122.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
A01_Split20000-1-2_patch_134.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
A01_Split20000-1-2_patch_136.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
A01_Split20000-1-2_patch_144.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
A01_Split20000-1-2_patch_149.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
A01_Split20000-1-2_patch_157.jpg	N/A	N/A	N/A	Treino	Treino	Teste	Treino
A01_Split20000-1-2_patch_160.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
A01_Split20000-1-2_patch_161.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
A01_Split20000-1-3_patch_000.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
A01_Split20000-1-3_patch_001.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
A01_Split20000-1-3_patch_040.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
A01_Split20000-1-3_patch_052.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
A01_Split20000-2-1_patch_001.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
A01_Split20000-2-1_patch_014.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
A01_Split20000-2-1_patch_015.jpg	N/A	N/A	Treino	Teste	Teste	Treino	Teste
A01_Split20000-2-1_patch_016.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
A01_Split20000-2-1_patch_026.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
A01_Split20000-2-1_patch_027.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
A01_Split20000-2-1_patch_066.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
A01_Split20000-2-1_patch_092.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
A01_Split20000-2-1_patch_104.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_008.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-0-0_patch_010.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_016.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_017.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-0-0_patch_018.jpg	N/A	N/A	N/A	N/A	Teste	Treino	Teste
Area02_30000-0-0_patch_019.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-0-0_patch_022.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Teste

Imagens	AL v. 1	AL v. 2	AL v. 3	AL v. 4	AL v. 5	Produção	Produção AL
Area02_30000-0-0_patch_025.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-0-0_patch_027.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-0-0_patch_039.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_040.jpg	N/A	N/A	N/A	N/A	N/A	Teste	Treino
Area02_30000-0-0_patch_048.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-0-0_patch_049.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-0-0_patch_050.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-0-0_patch_059.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_062.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-0-0_patch_067.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-0-0_patch_071.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_074.jpg	N/A	N/A	Treino	Treino	Teste	Treino	Teste
Area02_30000-0-0_patch_079.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_080.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_081.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-0-0_patch_084.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-0-0_patch_090.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_091.jpg	N/A	N/A	N/A	N/A	N/A	Teste	Treino
Area02_30000-0-0_patch_092.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-0-0_patch_094.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-0-0_patch_105.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_110.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Teste
Area02_30000-0-0_patch_115.jpg	N/A	N/A	N/A	N/A	Teste	Treino	Teste
Area02_30000-0-0_patch_131.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-0-0_patch_138.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-0-0_patch_147.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-0-0_patch_150.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-0-0_patch_151.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-0-0_patch_155.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_161.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Teste
Area02_30000-0-0_patch_164.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-0-0_patch_168.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-0-0_patch_169.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-0-0_patch_170.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_173.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_177.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_178.jpg	N/A	N/A	Teste	Teste	Teste	Treino	Teste
Area02_30000-0-0_patch_179.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_180.jpg	N/A	N/A	Treino	Treino	Treino	Teste	Treino
Area02_30000-0-0_patch_189.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_194.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-0-0_patch_196.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_205.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-0-0_patch_211.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-0-0_patch_216.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-0-0_patch_218.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_220.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-0-0_patch_221.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_222.jpg	N/A	N/A	Teste	Teste	Teste	Treino	Teste
Area02_30000-0-0_patch_223.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Teste
Area02_30000-0-0_patch_224.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-0-0_patch_233.jpg	N/A	N/A	N/A	N/A	N/A	Teste	Treino
Area02_30000-0-0_patch_249.jpg	N/A	N/A	Treino	Teste	Teste	Treino	Teste
Area02_30000-0-0_patch_255.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-0-0_patch_256.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-0-0_patch_265.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino

Imagens	AL v. 1	AL v. 2	AL v. 3	AL v. 4	AL v. 5	Produção	Produção AL
Area02_30000-0-0_patch_273.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-0-0_patch_275.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-0-0_patch_278.jpg	N/A	N/A	N/A	N/A	N/A	Teste	Treino
Area02_30000-0-0_patch_284.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_289.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-0-0_patch_324.jpg	N/A	N/A	N/A	Treino	Treino	Teste	Treino
Area02_30000-0-0_patch_325.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-0-0_patch_340.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-0-0_patch_351.jpg	N/A	N/A	N/A	Treino	Treino	Teste	Treino
Area02_30000-0-0_patch_361.jpg	N/A	N/A	N/A	Teste	Teste	Treino	Teste
Area02_30000-0-0_patch_362.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-0-0_patch_367.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_379.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_380.jpg	N/A	N/A	N/A	Treino	Treino	Teste	Treino
Area02_30000-0-0_patch_400.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_407.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_410.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-0-0_patch_436.jpg	N/A	N/A	Treino	Treino	Treino	N/A	N/A
Area02_30000-0-1_patch_000.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_001.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_003.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_004.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_012.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_023.jpg	N/A	N/A	N/A	N/A	Treino	Teste	Treino
Area02_30000-0-1_patch_024.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-0-1_patch_025.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-0-1_patch_031.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-0-1_patch_044.jpg	N/A	N/A	Treino	Treino	Treino	Teste	Treino
Area02_30000-0-1_patch_046.jpg	N/A	N/A	N/A	Teste	Teste	Treino	Teste
Area02_30000-0-1_patch_047.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_049.jpg	N/A	N/A	N/A	N/A	Teste	Treino	Teste
Area02_30000-0-1_patch_053.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-0-1_patch_056.jpg	N/A	N/A	Teste	Teste	Teste	Teste	Teste
Area02_30000-0-1_patch_064.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-0-1_patch_068.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_070.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_071.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_073.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-0-1_patch_074.jpg	N/A	N/A	N/A	N/A	Teste	Treino	Teste
Area02_30000-0-1_patch_078.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-0-1_patch_087.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_090.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-0-1_patch_108.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-0-1_patch_109.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_111.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_112.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_119.jpg	N/A	N/A	Teste	Teste	Teste	Treino	Teste
Area02_30000-0-1_patch_129.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_139.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_142.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-0-1_patch_149.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_150.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-0-1_patch_154.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-0-1_patch_156.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_158.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_159.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino

Imagens	AL v. 1	AL v. 2	AL v. 3	AL v. 4	AL v. 5	Produção	Produção AL
Area02_30000-0-1_patch_175.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-0-1_patch_176.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_177.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-0-1_patch_180.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_182.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_190.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-0-1_patch_191.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_192.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-0-1_patch_196.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-0-1_patch_197.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_200.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-0-1_patch_204.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-0-1_patch_213.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_215.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-0-1_patch_222.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-0-1_patch_224.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_231.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_235.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_239.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-0-1_patch_241.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_243.jpg	N/A	N/A	N/A	N/A	Treino	Teste	Treino
Area02_30000-0-1_patch_246.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_252.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-0-1_patch_253.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-0-1_patch_254.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-0-1_patch_256.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-0-1_patch_259.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-0-1_patch_263.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_264.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-0-1_patch_265.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-0-1_patch_274.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-0-1_patch_277.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_278.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-0-1_patch_279.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_300.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-0-1_patch_301.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_315.jpg	N/A	N/A	N/A	N/A	Teste	Treino	Teste
Area02_30000-0-1_patch_316.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-0-1_patch_317.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_344.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_345.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-0-1_patch_347.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-0-1_patch_358.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-0-1_patch_369.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_371.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-0-1_patch_372.jpg	N/A	N/A	Treino	Treino	Treino	Teste	Treino
Area02_30000-0-1_patch_380.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-0-1_patch_388.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_393.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_407.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-0-1_patch_411.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-1-0_patch_000.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_002.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_005.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_007.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_012.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino

Imagens	AL v. 1	AL v. 2	AL v. 3	AL v. 4	AL v. 5	Produção	Produção AL
Area02_30000-1-0_patch_016.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_021.jpg	N/A	N/A	N/A	N/A	Treino	Teste	Treino
Area02_30000-1-0_patch_023.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_028.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_029.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_033.jpg	N/A	N/A	N/A	Treino	Treino	Teste	Treino
Area02_30000-1-0_patch_035.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-1-0_patch_036.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_039.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-1-0_patch_047.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-1-0_patch_056.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-1-0_patch_058.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_059.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_067.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-1-0_patch_071.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_074.jpg	N/A	N/A	N/A	Treino	Treino	Teste	Treino
Area02_30000-1-0_patch_080.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-1-0_patch_081.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_084.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-1-0_patch_092.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-1-0_patch_097.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_107.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_114.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_121.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-1-0_patch_122.jpg	N/A	N/A	N/A	N/A	N/A	Teste	Treino
Area02_30000-1-0_patch_123.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_128.jpg	N/A	N/A	Teste	Teste	Teste	Treino	Teste
Area02_30000-1-0_patch_131.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-1-0_patch_133.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_137.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-1-0_patch_140.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-1-0_patch_142.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_145.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_147.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-1-0_patch_157.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Teste
Area02_30000-1-0_patch_159.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_164.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_170.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_175.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-1-0_patch_177.jpg	N/A	N/A	N/A	Treino	Treino	Teste	Treino
Area02_30000-1-0_patch_181.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-1-0_patch_183.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-1-0_patch_184.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-1-0_patch_197.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-1-0_patch_199.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_201.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-1-0_patch_210.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-1-0_patch_211.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-1-0_patch_220.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_224.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_227.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-1-0_patch_245.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-1-1_patch_001.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-1-1_patch_002.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-1-1_patch_004.jpg	N/A	N/A	N/A	N/A	N/A	Teste	Treino
Area02_30000-1-1_patch_006.jpg	N/A	N/A	N/A	N/A	Treino	Teste	Treino

Imagens	AL v. 1	AL v. 2	AL v. 3	AL v. 4	AL v. 5	Produção	Produção AL
Area02_30000-1-1_patch_027.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-1-1_patch_042.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-1_patch_047.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-1_patch_064.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-1-1_patch_065.jpg	N/A	N/A	Teste	Teste	Teste	Treino	Teste
Area02_30000-1-1_patch_067.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-1_patch_068.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Teste
Area02_30000-1-1_patch_069.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-1_patch_089.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
Area02_30000-1-1_patch_090.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
Area02_30000-1-1_patch_106.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
Area02_30000-1-1_patch_108.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
Area02_30000-1-1_patch_130.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-1-1_patch_149.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
Area02_30000-1-1_patch_151.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
Area02_30000-1-1_patch_152.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-1-1_patch_171.jpg	N/A	N/A	Treino	Treino	Treino	Teste	Teste
Area02_30000-1-1_patch_192.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-1-1_patch_211.jpg	N/A	N/A	N/A	Treino	Treino	Treino	Treino
Area02_30000-1-1_patch_231.jpg	N/A	N/A	N/A	N/A	N/A	Treino	Treino
Area02_30000-1-1_patch_232.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino
Area02_30000-1-1_patch_233.jpg	N/A	N/A	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-1_patch_274.jpg	N/A	N/A	N/A	N/A	Treino	Treino	Treino

Tabela C.2: Relação das imagens utilizadas durante o procedimento de validação cruzada *k-fold* da YOLOv4.

Imagens	*Fold* 01	*Fold* 02	*Fold* 03	*Fold* 04	*Fold* 05
A01_Split20000-0-0_patch_168.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_011.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_012.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_024.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_025.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_036.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_037.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_038.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_049.jpg	Treino	Teste	Treino	Treino	Treino
A01_Split20000-0-1_patch_050.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_051.jpg	Treino	Teste	Treino	Treino	Treino
A01_Split20000-0-1_patch_062.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_063.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_064.jpg	Treino	Teste	Treino	Treino	Treino
A01_Split20000-0-1_patch_075.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_076.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_077.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_086.jpg	Treino	Teste	Treino	Treino	Treino

Imagens	*Fold* 01	*Fold* 02	*Fold* 03	*Fold* 04	*Fold* 05
A01_Split20000-0-1_patch_087.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_088.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_089.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_090.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_099.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_100.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_101.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_102.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_103.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_111.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_112.jpg	Treino	Treino	Treino	Treino	Teste
A01_Split20000-0-1_patch_113.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_114.jpg	Treino	Teste	Treino	Treino	Treino
A01_Split20000-0-1_patch_115.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_116.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_125.jpg	Treino	Treino	Treino	Treino	Teste
A01_Split20000-0-1_patch_126.jpg	Treino	Teste	Treino	Treino	Treino
A01_Split20000-0-1_patch_127.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_128.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_129.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_138.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_139.jpg	Treino	Treino	Treino	Teste	Treino
A01_Split20000-0-1_patch_140.jpg	Treino	Teste	Treino	Treino	Treino
A01_Split20000-0-1_patch_141.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_142.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_150.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_151.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_152.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_153.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_154.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_155.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_163.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_164.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_165.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_166.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_167.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_168.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_006.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_007.jpg	Treino	Treino	Treino	Treino	Treino

Imagens	*Fold* 01	*Fold* 02	*Fold* 03	*Fold* 04	*Fold* 05
A01_Split20000-0-2_patch_008.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_009.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_010.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_011.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_012.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_019.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_020.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_021.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_022.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_023.jpg	Treino	Treino	Teste	Treino	Treino
A01_Split20000-0-2_patch_024.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_025.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_030.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_031.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_032.jpg	Treino	Treino	Treino	Teste	Treino
A01_Split20000-0-2_patch_033.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_034.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_035.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_036.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_037.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_038.jpg	Treino	Teste	Treino	Treino	Treino
A01_Split20000-0-2_patch_043.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_044.jpg	Treino	Teste	Treino	Treino	Treino
A01_Split20000-0-2_patch_045.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_046.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_047.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_048.jpg	Teste	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_049.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_050.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_051.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_055.jpg	Treino	Teste	Treino	Treino	Treino
A01_Split20000-0-2_patch_056.jpg	Teste	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_057.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_058.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_059.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_060.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_061.jpg	Treino	Teste	Treino	Treino	Treino
A01_Split20000-0-2_patch_062.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_063.jpg	Treino	Treino	Treino	Treino	Treino

Imagens	*Fold* 01	*Fold* 02	*Fold* 03	*Fold* 04	*Fold* 05
A01_Split20000-0-2_patch_064.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_068.jpg	Treino	Treino	Treino	Teste	Treino
A01_Split20000-0-2_patch_069.jpg	Teste	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_070.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_071.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_072.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_073.jpg	Treino	Teste	Treino	Treino	Treino
A01_Split20000-0-2_patch_074.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_075.jpg	Treino	Treino	Treino	Teste	Treino
A01_Split20000-0-2_patch_076.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_077.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_081.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_082.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_083.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_084.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_085.jpg	Treino	Teste	Treino	Treino	Treino
A01_Split20000-0-2_patch_086.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_088.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_089.jpg	Treino	Treino	Teste	Treino	Treino
A01_Split20000-0-2_patch_090.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_094.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_098.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_100.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_102.jpg	Treino	Treino	Teste	Treino	Treino
A01_Split20000-0-2_patch_106.jpg	Treino	Treino	Treino	Teste	Treino
A01_Split20000-0-2_patch_126.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_128.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_137.jpg	Treino	Treino	Treino	Teste	Treino
A01_Split20000-0-2_patch_138.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_145.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_146.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_147.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_153.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_158.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_159.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_162.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_163.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_164.jpg	Treino	Teste	Treino	Treino	Treino
A01_Split20000-0-2_patch_166.jpg	Teste	Treino	Treino	Treino	Treino

Imagens	*Fold* 01	*Fold* 02	*Fold* 03	*Fold* 04	*Fold* 05
A01_Split20000-0-3_patch_025.jpg	Treino	Treino	Teste	Treino	Treino
A01_Split20000-0-3_patch_038.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-3_patch_051.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_017.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_031.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_043.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_045.jpg	Treino	Treino	Treino	Treino	Teste
A01_Split20000-1-0_patch_046.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_057.jpg	Teste	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_058.jpg	Treino	Treino	Teste	Treino	Treino
A01_Split20000-1-0_patch_060.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_067.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_073.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_080.jpg	Treino	Treino	Treino	Treino	Teste
A01_Split20000-1-0_patch_084.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_086.jpg	Treino	Treino	Teste	Treino	Treino
A01_Split20000-1-0_patch_094.jpg	Treino	Treino	Treino	Teste	Treino
A01_Split20000-1-0_patch_095.jpg	Treino	Treino	Treino	Treino	Teste
A01_Split20000-1-0_patch_096.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_097.jpg	Teste	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_102.jpg	Treino	Treino	Treino	Teste	Treino
A01_Split20000-1-0_patch_112.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_113.jpg	Treino	Treino	Teste	Treino	Treino
A01_Split20000-1-0_patch_118.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_121.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_126.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_127.jpg	Treino	Treino	Treino	Teste	Treino
A01_Split20000-1-0_patch_130.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_131.jpg	Treino	Teste	Treino	Treino	Treino
A01_Split20000-1-0_patch_132.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_142.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_144.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_153.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_164.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_165.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_168.jpg	Treino	Treino	Treino	Teste	Treino
A01_Split20000-1-1_patch_006.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_011.jpg	Treino	Treino	Treino	Treino	Teste
A01_Split20000-1-1_patch_012.jpg	Treino	Treino	Treino	Treino	Treino

Imagens	*Fold* 01	*Fold* 02	*Fold* 03	*Fold* 04	*Fold* 05
A01_Split20000-1-1_patch_013.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_022.jpg	Teste	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_024.jpg	Treino	Teste	Treino	Treino	Treino
A01_Split20000-1-1_patch_031.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_035.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_037.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_044.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_045.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_046.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_048.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_055.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_058.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_060.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_061.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_063.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_072.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_073.jpg	Treino	Treino	Treino	Teste	Treino
A01_Split20000-1-1_patch_082.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_085.jpg	Treino	Teste	Treino	Treino	Treino
A01_Split20000-1-1_patch_100.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_104.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_105.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_108.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_118.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_122.jpg	Treino	Treino	Teste	Treino	Treino
A01_Split20000-1-1_patch_125.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_127.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_133.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_136.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_142.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_147.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_156.jpg	Treino	Treino	Teste	Treino	Treino
A01_Split20000-1-1_patch_157.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-2_patch_003.jpg	Teste	Treino	Treino	Treino	Treino
A01_Split20000-1-2_patch_008.jpg	N/A	N/A	N/A	N/A	N/A
A01_Split20000-1-2_patch_009.jpg	Teste	Treino	Treino	Treino	Treino
A01_Split20000-1-2_patch_016.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-2_patch_021.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-2_patch_030.jpg	Treino	Treino	Treino	Treino	Treino

Imagens	*Fold* 01	*Fold* 02	*Fold* 03	*Fold* 04	*Fold* 05
A01_Split20000-1-2_patch_046.jpg	Treino	Treino	Teste	Treino	Treino
A01_Split20000-1-2_patch_047.jpg	Treino	Teste	Treino	Treino	Treino
A01_Split20000-1-2_patch_053.jpg	Treino	Treino	Treino	Treino	Teste
A01_Split20000-1-2_patch_057.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-2_patch_058.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-2_patch_059.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-2_patch_074.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-2_patch_084.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-2_patch_085.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-2_patch_086.jpg	Treino	Treino	Treino	Teste	Treino
A01_Split20000-1-2_patch_092.jpg	Treino	Treino	Treino	Treino	Teste
A01_Split20000-1-2_patch_099.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-2_patch_106.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-2_patch_110.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-2_patch_122.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-2_patch_134.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-2_patch_136.jpg	Treino	Teste	Treino	Treino	Treino
A01_Split20000-1-2_patch_144.jpg	Treino	Treino	Treino	Teste	Treino
A01_Split20000-1-2_patch_149.jpg	Teste	Treino	Treino	Treino	Treino
A01_Split20000-1-2_patch_157.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-2_patch_160.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-2_patch_161.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-3_patch_000.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-3_patch_001.jpg	Treino	Treino	Treino	Treino	Teste
A01_Split20000-1-3_patch_040.jpg	Treino	Treino	Treino	Treino	Teste
A01_Split20000-1-3_patch_052.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-2-1_patch_001.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-2-1_patch_014.jpg	Treino	Treino	Treino	Treino	Teste
A01_Split20000-2-1_patch_015.jpg	Teste	Treino	Treino	Treino	Treino
A01_Split20000-2-1_patch_016.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-2-1_patch_026.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-2-1_patch_027.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-2-1_patch_066.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-2-1_patch_092.jpg	Treino	Treino	Treino	Treino	Treino
A01_Split20000-2-1_patch_104.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_008.jpg	Treino	Treino	Treino	Treino	Teste
Area02_30000-0-0_patch_010.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_016.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_017.jpg	Treino	Treino	Treino	Teste	Treino

Imagens	*Fold* 01	*Fold* 02	*Fold* 03	*Fold* 04	*Fold* 05
Area02_30000-0-0_patch_018.jpg	Treino	Treino	Teste	Treino	Treino
Area02_30000-0-0_patch_019.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_022.jpg	Treino	Treino	Treino	Teste	Treino
Area02_30000-0-0_patch_025.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_027.jpg	Treino	Treino	Treino	Treino	Teste
Area02_30000-0-0_patch_039.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_040.jpg	Treino	Treino	Treino	Treino	Teste
Area02_30000-0-0_patch_048.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_049.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_050.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_059.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_062.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_067.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_071.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_074.jpg	Treino	Treino	Treino	Teste	Treino
Area02_30000-0-0_patch_079.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_080.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_081.jpg	Treino	Teste	Treino	Treino	Treino
Area02_30000-0-0_patch_084.jpg	Teste	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_090.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_091.jpg	Treino	Treino	Teste	Treino	Treino
Area02_30000-0-0_patch_092.jpg	Teste	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_094.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_105.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_110.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_115.jpg	Treino	Treino	Treino	Treino	Teste
Area02_30000-0-0_patch_131.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_138.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_147.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_150.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_151.jpg	Treino	Treino	Teste	Treino	Treino
Area02_30000-0-0_patch_155.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_161.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_164.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_168.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_169.jpg	Treino	Treino	Treino	Teste	Treino
Area02_30000-0-0_patch_170.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_173.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_177.jpg	Treino	Treino	Treino	Treino	Treino

Imagens	*Fold* 01	*Fold* 02	*Fold* 03	*Fold* 04	*Fold* 05
Area02_30000-0-0_patch_178.jpg	Treino	Treino	Teste	Treino	Treino
Area02_30000-0-0_patch_179.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_180.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_189.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_194.jpg	Treino	Treino	Treino	Treino	Teste
Area02_30000-0-0_patch_196.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_205.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_211.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_216.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_218.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_220.jpg	Treino	Treino	Treino	Teste	Treino
Area02_30000-0-0_patch_221.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_222.jpg	Teste	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_223.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_224.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_233.jpg	Treino	Teste	Treino	Treino	Treino
Area02_30000-0-0_patch_249.jpg	Treino	Treino	Teste	Treino	Treino
Area02_30000-0-0_patch_255.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_256.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_265.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_273.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_275.jpg	Teste	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_278.jpg	Treino	Teste	Treino	Treino	Treino
Area02_30000-0-0_patch_284.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_289.jpg	Treino	Treino	Treino	Treino	Teste
Area02_30000-0-0_patch_324.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_325.jpg	Treino	Treino	Teste	Treino	Treino
Area02_30000-0-0_patch_340.jpg	Teste	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_351.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_361.jpg	Treino	Treino	Treino	Treino	Teste
Area02_30000-0-0_patch_362.jpg	Treino	Treino	Treino	Treino	Teste
Area02_30000-0-0_patch_367.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_379.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_380.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_400.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_407.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_410.jpg	Treino	Treino	Treino	Treino	Teste
Area02_30000-0-0_patch_436.jpg	N/A	N/A	N/A	N/A	N/A
Area02_30000-0-1_patch_000.jpg	Treino	Treino	Treino	Treino	Treino

Imagens	*Fold* 01	*Fold* 02	*Fold* 03	*Fold* 04	*Fold* 05
Area02_30000-0-1_patch_001.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_003.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_004.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_012.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_023.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_024.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_025.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_031.jpg	Treino	Treino	Teste	Treino	Treino
Area02_30000-0-1_patch_044.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_046.jpg	Treino	Treino	Treino	Teste	Treino
Area02_30000-0-1_patch_047.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_049.jpg	Treino	Treino	Teste	Treino	Treino
Area02_30000-0-1_patch_053.jpg	Teste	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_056.jpg	Teste	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_064.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_068.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_070.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_071.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_073.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_074.jpg	Treino	Treino	Treino	Teste	Treino
Area02_30000-0-1_patch_078.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_087.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_090.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_108.jpg	Teste	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_109.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_111.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_112.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_119.jpg	Teste	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_129.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_139.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_142.jpg	Teste	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_149.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_150.jpg	Teste	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_154.jpg	Treino	Treino	Treino	Treino	Teste
Area02_30000-0-1_patch_156.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_158.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_159.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_175.jpg	Treino	Treino	Teste	Treino	Treino
Area02_30000-0-1_patch_176.jpg	Treino	Treino	Treino	Treino	Treino

Imagens	*Fold* 01	*Fold* 02	*Fold* 03	*Fold* 04	*Fold* 05
Area02_30000-0-1_patch_177.jpg	Treino	Treino	Treino	Teste	Treino
Area02_30000-0-1_patch_180.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_182.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_190.jpg	Treino	Treino	Teste	Treino	Treino
Area02_30000-0-1_patch_191.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_192.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_196.jpg	Treino	Treino	Treino	Teste	Treino
Area02_30000-0-1_patch_197.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_200.jpg	Treino	Treino	Treino	Treino	Teste
Area02_30000-0-1_patch_204.jpg	Treino	Treino	Teste	Treino	Treino
Area02_30000-0-1_patch_213.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_215.jpg	Treino	Treino	Treino	Treino	Teste
Area02_30000-0-1_patch_222.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_224.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_231.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_235.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_239.jpg	Treino	Teste	Treino	Treino	Treino
Area02_30000-0-1_patch_241.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_243.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_246.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_252.jpg	Treino	Treino	Treino	Treino	Teste
Area02_30000-0-1_patch_253.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_254.jpg	Teste	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_256.jpg	Treino	Treino	Treino	Treino	Teste
Area02_30000-0-1_patch_259.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_263.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_264.jpg	Treino	Teste	Treino	Treino	Treino
Area02_30000-0-1_patch_265.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_274.jpg	Treino	Teste	Treino	Treino	Treino
Area02_30000-0-1_patch_277.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_278.jpg	Treino	Treino	Treino	Teste	Treino
Area02_30000-0-1_patch_279.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_300.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_301.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_315.jpg	Teste	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_316.jpg	Treino	Treino	Teste	Treino	Treino
Area02_30000-0-1_patch_317.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_344.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_345.jpg	Treino	Treino	Treino	Treino	Treino

Imagens	*Fold* 01	*Fold* 02	*Fold* 03	*Fold* 04	*Fold* 05
Area02_30000-0-1_patch_347.jpg	Teste	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_358.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_369.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_371.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_372.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_380.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_388.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_393.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-1_patch_407.jpg	Treino	Treino	Treino	Treino	Teste
Area02_30000-0-1_patch_411.jpg	Treino	Treino	Teste	Treino	Treino
Area02_30000-1-0_patch_000.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_002.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_005.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_007.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_012.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_016.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_021.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_023.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_028.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_029.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_033.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_035.jpg	Treino	Treino	Teste	Treino	Treino
Area02_30000-1-0_patch_036.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_039.jpg	Treino	Treino	Treino	Teste	Treino
Area02_30000-1-0_patch_047.jpg	Teste	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_056.jpg	Treino	Treino	Treino	Teste	Treino
Area02_30000-1-0_patch_058.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_059.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_067.jpg	Treino	Teste	Treino	Treino	Treino
Area02_30000-1-0_patch_071.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_074.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_080.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_081.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_084.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_092.jpg	Treino	Treino	Teste	Treino	Treino
Area02_30000-1-0_patch_097.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_107.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_114.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_121.jpg	Treino	Treino	Treino	Treino	Treino

Imagens	*Fold* 01	*Fold* 02	*Fold* 03	*Fold* 04	*Fold* 05
Area02_30000-1-0_patch_122.jpg	Treino	Teste	Treino	Treino	Treino
Area02_30000-1-0_patch_123.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_128.jpg	Treino	Treino	Treino	Teste	Treino
Area02_30000-1-0_patch_131.jpg	Teste	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_133.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_137.jpg	Teste	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_140.jpg	Treino	Teste	Treino	Treino	Treino
Area02_30000-1-0_patch_142.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_145.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_147.jpg	Treino	Teste	Treino	Treino	Treino
Area02_30000-1-0_patch_157.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_159.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_164.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_170.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_175.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_177.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_181.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_183.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_184.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_197.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_199.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_201.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_210.jpg	Treino	Treino	Treino	Teste	Treino
Area02_30000-1-0_patch_211.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_220.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_224.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-0_patch_227.jpg	Treino	Treino	Treino	Treino	Teste
Area02_30000-1-0_patch_245.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-1_patch_001.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-1_patch_002.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-1_patch_004.jpg	Treino	Treino	Treino	Teste	Treino
Area02_30000-1-1_patch_006.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-1_patch_027.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-1_patch_042.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-1_patch_047.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-1_patch_064.jpg	Treino	Treino	Teste	Treino	Treino
Area02_30000-1-1_patch_065.jpg	Treino	Treino	Treino	Treino	Teste
Area02_30000-1-1_patch_067.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-1_patch_068.jpg	Treino	Treino	Treino	Treino	Treino

Imagens	*Fold* 01	*Fold* 02	*Fold* 03	*Fold* 04	*Fold* 05
Area02_30000-1-1_patch_069.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-1_patch_089.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-1_patch_090.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-1_patch_106.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-1_patch_108.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-1_patch_130.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-1_patch_149.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-1_patch_151.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-1_patch_152.jpg	Treino	Treino	Teste	Treino	Treino
Area02_30000-1-1_patch_171.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-1_patch_192.jpg	Treino	Treino	Teste	Treino	Treino
Area02_30000-1-1_patch_211.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-1_patch_231.jpg	Teste	Treino	Treino	Treino	Treino
Area02_30000-1-1_patch_232.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-1_patch_233.jpg	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-1_patch_274.jpg	Treino	Treino	Treino	Treino	Treino

Tabela C.3: Relação das imagens utilizadas durante o procedimento de validação cruzada *k-fold* da YOLOv9.

Imagens	*Fold* 01	*Fold* 02	*Fold* 03	*Fold* 04	*Fold* 05	*Fold* 06	*Fold* 07	*Fold* 08	*Fold* 09	*Fold* 10
A01_Split20000-0-0_patch_168.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino
A01_Split20000-0-1_patch_011.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino
A01_Split20000-0-1_patch_012.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_024.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste
A01_Split20000-0-1_patch_025.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino
A01_Split20000-0-1_patch_036.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino
A01_Split20000-0-1_patch_037.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste
A01_Split20000-0-1_patch_038.jpg	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_049.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino
A01_Split20000-0-1_patch_050.jpg	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_051.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino
A01_Split20000-0-1_patch_062.jpg	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_063.jpg	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_064.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste
A01_Split20000-0-1_patch_075.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste
A01_Split20000-0-1_patch_076.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino
A01_Split20000-0-1_patch_077.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino
A01_Split20000-0-1_patch_086.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste
A01_Split20000-0-1_patch_087.jpg	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_088.jpg	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_089.jpg	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_090.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino
A01_Split20000-0-1_patch_099.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino
A01_Split20000-0-1_patch_100.jpg	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_101.jpg	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_102.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino
A01_Split20000-0-1_patch_103.jpg	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_111.jpg	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_112.jpg	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_113.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_114.jpg	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_115.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino
A01_Split20000-0-1_patch_116.jpg	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_125.jpg	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_126.jpg	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_127.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino
A01_Split20000-0-1_patch_128.jpg	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_129.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino
A01_Split20000-0-1_patch_138.jpg	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino

Imagens	*Fold* 01	*Fold* 02	*Fold* 03	*Fold* 04	*Fold* 05	*Fold* 06	*Fold* 07	*Fold* 08	*Fold* 09	*Fold* 10
A01_Split20000-0-1_patch_139.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino
A01_Split20000-0-1_patch_140.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino
A01_Split20000-0-1_patch_141.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_142.jpg	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_150.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino
A01_Split20000-0-1_patch_151.jpg	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_152.jpg	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_153.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino
A01_Split20000-0-1_patch_154.jpg	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_155.jpg	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_163.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_164.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste
A01_Split20000-0-1_patch_165.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste
A01_Split20000-0-1_patch_166.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino
A01_Split20000-0-1_patch_167.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino
A01_Split20000-0-1_patch_168.jpg	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_006.jpg	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_007.jpg	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_008.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino
A01_Split20000-0-2_patch_009.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste
A01_Split20000-0-2_patch_010.jpg	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_011.jpg	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_012.jpg	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_019.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste
A01_Split20000-0-2_patch_020.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_021.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_022.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino
A01_Split20000-0-2_patch_023.jpg	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_024.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino
A01_Split20000-0-2_patch_025.jpg	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_030.jpg	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_031.jpg	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_032.jpg	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_033.jpg	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_034.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino
A01_Split20000-0-2_patch_035.jpg	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_036.jpg	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_037.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino
A01_Split20000-0-2_patch_038.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_043.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino
A01_Split20000-0-2_patch_044.jpg	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_045.jpg	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_046.jpg	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_047.jpg	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_048.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino
A01_Split20000-0-2_patch_049.jpg	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_050.jpg	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_051.jpg	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_055.jpg	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_056.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino
A01_Split20000-0-2_patch_057.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste
A01_Split20000-0-2_patch_058.jpg	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_059.jpg	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_060.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino
A01_Split20000-0-2_patch_061.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino
A01_Split20000-0-2_patch_062.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_063.jpg	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_064.jpg	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_068.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste
A01_Split20000-0-2_patch_069.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_070.jpg	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_071.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino
A01_Split20000-0-2_patch_072.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste
A01_Split20000-0-2_patch_073.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste
A01_Split20000-0-2_patch_074.jpg	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_075.jpg	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_076.jpg	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_077.jpg	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_081.jpg	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_082.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino
A01_Split20000-0-2_patch_083.jpg	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_084.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste
A01_Split20000-0-2_patch_085.jpg	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_086.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste
A01_Split20000-0-2_patch_088.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino
A01_Split20000-0-2_patch_089.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino

Imagens	*Fold* 01	*Fold* 02	*Fold* 03	*Fold* 04	*Fold* 05	*Fold* 06	*Fold* 07	*Fold* 08	*Fold* 09	*Fold* 10
A01_Split20000-0-2_patch_090.jpg	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_094.jpg	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_098.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste
A01_Split20000-0-2_patch_100.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste
A01_Split20000-0-2_patch_102.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino
A01_Split20000-0-2_patch_106.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste
A01_Split20000-0-2_patch_126.jpg	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_128.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino
A01_Split20000-0-2_patch_137.jpg	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_138.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_145.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_146.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino
A01_Split20000-0-2_patch_147.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste
A01_Split20000-0-2_patch_153.jpg	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_158.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_159.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino
A01_Split20000-0-2_patch_162.jpg	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-2_patch_163.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino
A01_Split20000-0-2_patch_164.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino
A01_Split20000-0-2_patch_166.jpg	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-3_patch_025.jpg	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-0-3_patch_038.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste
A01_Split20000-0-3_patch_051.jpg	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_017.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste
A01_Split20000-1-0_patch_031.jpg	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_043.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino
A01_Split20000-1-0_patch_045.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino
A01_Split20000-1-0_patch_046.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_057.jpg	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_058.jpg	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_060.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino
A01_Split20000-1-0_patch_067.jpg	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_073.jpg	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_080.jpg	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_084.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_086.jpg	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_094.jpg	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_095.jpg	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_096.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino
A01_Split20000-1-0_patch_097.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino
A01_Split20000-1-0_patch_102.jpg	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_112.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino
A01_Split20000-1-0_patch_113.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino
A01_Split20000-1-0_patch_118.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_121.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino
A01_Split20000-1-0_patch_126.jpg	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_127.jpg	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_130.jpg	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_131.jpg	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_132.jpg	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_142.jpg	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_144.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste
A01_Split20000-1-0_patch_153.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste
A01_Split20000-1-0_patch_164.jpg	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_165.jpg	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-0_patch_168.jpg	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_006.jpg	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_011.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino
A01_Split20000-1-1_patch_012.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_013.jpg	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_022.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino
A01_Split20000-1-1_patch_024.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino
A01_Split20000-1-1_patch_031.jpg	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_035.jpg	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_037.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste
A01_Split20000-1-1_patch_044.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino
A01_Split20000-1-1_patch_045.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_046.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino
A01_Split20000-1-1_patch_048.jpg	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_055.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino
A01_Split20000-1-1_patch_058.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino
A01_Split20000-1-1_patch_060.jpg	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_061.jpg	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_063.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino
A01_Split20000-1-1_patch_072.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_073.jpg	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino

Imagens	*Fold* 01	*Fold* 02	*Fold* 03	*Fold* 04	*Fold* 05	*Fold* 06	*Fold* 07	*Fold* 08	*Fold* 09	*Fold* 10
A01_Split20000-1-1_patch_082.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino
A01_Split20000-1-1_patch_085.jpg	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_100.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino
A01_Split20000-1-1_patch_104.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino
A01_Split20000-1-1_patch_105.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino
A01_Split20000-1-1_patch_108.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino
A01_Split20000-1-1_patch_118.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino
A01_Split20000-1-1_patch_122.jpg	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_125.jpg	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_127.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino
A01_Split20000-1-1_patch_133.jpg	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_136.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino
A01_Split20000-1-1_patch_142.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino
A01_Split20000-1-1_patch_147.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste
A01_Split20000-1-1_patch_156.jpg	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-1_patch_157.jpg	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-2_patch_003.jpg	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-2_patch_008.jpg	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
A01_Split20000-1-2_patch_009.jpg	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-2_patch_016.jpg	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-2_patch_021.jpg	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-2_patch_030.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste
A01_Split20000-1-2_patch_046.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino
A01_Split20000-1-2_patch_047.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino
A01_Split20000-1-2_patch_053.jpg	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-2_patch_057.jpg	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-2_patch_058.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino
A01_Split20000-1-2_patch_059.jpg	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-2_patch_074.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino
A01_Split20000-1-2_patch_084.jpg	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-2_patch_085.jpg	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-2_patch_086.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste
A01_Split20000-1-2_patch_092.jpg	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-2_patch_099.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino
A01_Split20000-1-2_patch_106.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino
A01_Split20000-1-2_patch_110.jpg	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-2_patch_122.jpg	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-2_patch_134.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste
A01_Split20000-1-2_patch_136.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino
A01_Split20000-1-2_patch_144.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino
A01_Split20000-1-2_patch_149.jpg	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-2_patch_157.jpg	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-2_patch_160.jpg	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-1-2_patch_161.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino
A01_Split20000-1-3_patch_000.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino
A01_Split20000-1-3_patch_001.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste
A01_Split20000-1-3_patch_040.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino
A01_Split20000-1-3_patch_052.jpg	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-2-1_patch_001.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste
A01_Split20000-2-1_patch_014.jpg	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-2-1_patch_015.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino
A01_Split20000-2-1_patch_016.jpg	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-2-1_patch_026.jpg	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
A01_Split20000-2-1_patch_027.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino
A01_Split20000-2-1_patch_066.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino
A01_Split20000-2-1_patch_092.jpg	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino
A01_Split20000-2-1_patch_104.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino
Area02_30000-0-0_patch_008.jpg	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_010.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_016.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_017.jpg	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_018.jpg	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_019.jpg	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_022.jpg	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_025.jpg	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_027.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino
Area02_30000-0-0_patch_039.jpg	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_040.jpg	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_048.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino
Area02_30000-0-0_patch_049.jpg	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_050.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino
Area02_30000-0-0_patch_059.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_062.jpg	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02_30000-0-0_patch_067.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste
Area02_30000-0-0_patch_071.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino
Area02_30000-0-0_patch_074.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino

Imagens	*Fold* 01	*Fold* 02	*Fold* 03	*Fold* 04	*Fold* 05	*Fold* 06	*Fold* 07	*Fold* 08	*Fold* 09	*Fold* 10
Area02__30000-0-0__patch__079.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino
Area02__30000-0-0__patch__080.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino
Area02__30000-0-0__patch__081.jpg	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-0__patch__084.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino
Area02__30000-0-0__patch__090.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino
Area02__30000-0-0__patch__091.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino
Area02__30000-0-0__patch__092.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste
Area02__30000-0-0__patch__094.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino
Area02__30000-0-0__patch__105.jpg	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-0__patch__110.jpg	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-0__patch__115.jpg	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-0__patch__131.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino
Area02__30000-0-0__patch__138.jpg	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-0__patch__147.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino
Area02__30000-0-0__patch__150.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino
Area02__30000-0-0__patch__151.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino
Area02__30000-0-0__patch__155.jpg	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-0__patch__161.jpg	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-0__patch__164.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino
Area02__30000-0-0__patch__168.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino
Area02__30000-0-0__patch__169.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste
Area02__30000-0-0__patch__170.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino
Area02__30000-0-0__patch__173.jpg	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-0__patch__177.jpg	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-0__patch__178.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste
Area02__30000-0-0__patch__179.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino
Area02__30000-0-0__patch__180.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino
Area02__30000-0-0__patch__189.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino
Area02__30000-0-0__patch__194.jpg	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-0__patch__196.jpg	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-0__patch__205.jpg	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-0__patch__211.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino
Area02__30000-0-0__patch__216.jpg	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-0__patch__218.jpg	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-0__patch__220.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino
Area02__30000-0-0__patch__221.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino
Area02__30000-0-0__patch__222.jpg	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-0__patch__223.jpg	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-0__patch__224.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino
Area02__30000-0-0__patch__233.jpg	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-0__patch__249.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino
Area02__30000-0-0__patch__255.jpg	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-0__patch__256.jpg	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-0__patch__265.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino
Area02__30000-0-0__patch__273.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino
Area02__30000-0-0__patch__275.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino
Area02__30000-0-0__patch__278.jpg	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-0__patch__284.jpg	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-0__patch__289.jpg	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-0__patch__324.jpg	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-0__patch__325.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste
Area02__30000-0-0__patch__340.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino
Area02__30000-0-0__patch__351.jpg	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-0__patch__361.jpg	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-0__patch__362.jpg	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-0__patch__367.jpg	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-0__patch__379.jpg	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-0__patch__380.jpg	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-0__patch__400.jpg	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-0__patch__407.jpg	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-0__patch__410.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino
Area02__30000-0-0__patch__436.jpg	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Area02__30000-0-1__patch__000.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino
Area02__30000-0-1__patch__001.jpg	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__003.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste
Area02__30000-0-1__patch__004.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino
Area02__30000-0-1__patch__012.jpg	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__023.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino
Area02__30000-0-1__patch__024.jpg	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__025.jpg	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__031.jpg	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__044.jpg	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__046.jpg	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__047.jpg	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__049.jpg	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__053.jpg	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino

Imagens	*Fold* 01	*Fold* 02	*Fold* 03	*Fold* 04	*Fold* 05	*Fold* 06	*Fold* 07	*Fold* 08	*Fold* 09	*Fold* 10
Area02__30000-0-1__patch__056.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste
Area02__30000-0-1__patch__064.jpg	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__068.jpg	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__070.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino
Area02__30000-0-1__patch__071.jpg	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__073.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__074.jpg	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__078.jpg	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__087.jpg	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__090.jpg	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__108.jpg	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__109.jpg	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__111.jpg	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__112.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__119.jpg	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__129.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino
Area02__30000-0-1__patch__139.jpg	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__142.jpg	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__149.jpg	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__150.jpg	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__154.jpg	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__156.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino
Area02__30000-0-1__patch__158.jpg	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__159.jpg	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__175.jpg	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__176.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__177.jpg	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__180.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste
Area02__30000-0-1__patch__182.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino
Area02__30000-0-1__patch__190.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino
Area02__30000-0-1__patch__191.jpg	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__192.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino
Area02__30000-0-1__patch__196.jpg	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__197.jpg	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__200.jpg	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__204.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino
Area02__30000-0-1__patch__213.jpg	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__215.jpg	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__222.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino
Area02__30000-0-1__patch__224.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__231.jpg	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__235.jpg	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__239.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino
Area02__30000-0-1__patch__241.jpg	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__243.jpg	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__246.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste
Area02__30000-0-1__patch__252.jpg	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__253.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__254.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__256.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste
Area02__30000-0-1__patch__259.jpg	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__263.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste
Area02__30000-0-1__patch__264.jpg	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__265.jpg	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__274.jpg	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__277.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino
Area02__30000-0-1__patch__278.jpg	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__279.jpg	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__300.jpg	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__301.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino
Area02__30000-0-1__patch__315.jpg	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__316.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino
Area02__30000-0-1__patch__317.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino
Area02__30000-0-1__patch__344.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste
Area02__30000-0-1__patch__345.jpg	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__347.jpg	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__358.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino
Area02__30000-0-1__patch__369.jpg	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__371.jpg	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__372.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino
Area02__30000-0-1__patch__380.jpg	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__388.jpg	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__393.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino
Area02__30000-0-1__patch__407.jpg	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-0-1__patch__411.jpg	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-1-0__patch__000.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino

Imagens	*Fold* 01	*Fold* 02	*Fold* 03	*Fold* 04	*Fold* 05	*Fold* 06	*Fold* 07	*Fold* 08	*Fold* 09	*Fold* 10
Area02__30000-1-0__patch__002.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino
Area02__30000-1-0__patch__005.jpg	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-1-0__patch__007.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino
Area02__30000-1-0__patch__012.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino
Area02__30000-1-0__patch__016.jpg	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-1-0__patch__021.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino
Area02__30000-1-0__patch__023.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino
Area02__30000-1-0__patch__028.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste
Area02__30000-1-0__patch__029.jpg	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-1-0__patch__033.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino
Area02__30000-1-0__patch__035.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino
Area02__30000-1-0__patch__036.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino
Area02__30000-1-0__patch__039.jpg	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-1-0__patch__047.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino
Area02__30000-1-0__patch__056.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino
Area02__30000-1-0__patch__058.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino
Area02__30000-1-0__patch__059.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino
Area02__30000-1-0__patch__067.jpg	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino
Area02__30000-1-0__patch__071.jpg	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino
Area02__30000-1-0__patch__074.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino
Area02__30000-1-0__patch__080.jpg	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-1-0__patch__081.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino
Area02__30000-1-0__patch__084.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino
Area02__30000-1-0__patch__092.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino
Area02__30000-1-0__patch__097.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino
Area02__30000-1-0__patch__107.jpg	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-1-0__patch__114.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste
Area02__30000-1-0__patch__121.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino
Area02__30000-1-0__patch__122.jpg	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino
Area02__30000-1-0__patch__123.jpg	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-1-0__patch__128.jpg	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-1-0__patch__131.jpg	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-1-0__patch__133.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino
Area02__30000-1-0__patch__137.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste
Area02__30000-1-0__patch__140.jpg	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-1-0__patch__142.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste
Area02__30000-1-0__patch__145.jpg	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-1-0__patch__147.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino
Area02__30000-1-0__patch__157.jpg	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-1-0__patch__159.jpg	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-1-0__patch__164.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino
Area02__30000-1-0__patch__170.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino
Area02__30000-1-0__patch__175.jpg	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-1-0__patch__177.jpg	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-1-0__patch__181.jpg	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino
Area02__30000-1-0__patch__183.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino
Area02__30000-1-0__patch__184.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino
Area02__30000-1-0__patch__197.jpg	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino
Area02__30000-1-0__patch__199.jpg	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-1-0__patch__201.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino
Area02__30000-1-0__patch__210.jpg	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-1-0__patch__211.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino
Area02__30000-1-0__patch__220.jpg	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-1-0__patch__224.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino
Area02__30000-1-0__patch__227.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino
Area02__30000-1-0__patch__245.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino
Area02__30000-1-1__patch__001.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino
Area02__30000-1-1__patch__002.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste
Area02__30000-1-1__patch__004.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste
Area02__30000-1-1__patch__006.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino
Area02__30000-1-1__patch__027.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino
Area02__30000-1-1__patch__042.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino
Area02__30000-1-1__patch__047.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino
Area02__30000-1-1__patch__064.jpg	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-1-1__patch__065.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino
Area02__30000-1-1__patch__067.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino
Area02__30000-1-1__patch__068.jpg	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-1-1__patch__069.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino
Area02__30000-1-1__patch__089.jpg	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino
Area02__30000-1-1__patch__090.jpg	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-1-1__patch__106.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste
Area02__30000-1-1__patch__108.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino
Area02__30000-1-1__patch__130.jpg	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino
Area02__30000-1-1__patch__149.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino
Area02__30000-1-1__patch__151.jpg	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02__30000-1-1__patch__152.jpg	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino

Imagens	*Fold* 01	*Fold* 02	*Fold* 03	*Fold* 04	*Fold* 05	*Fold* 06	*Fold* 07	*Fold* 08	*Fold* 09	*Fold* 10
Area02_30000-1-1_patch_171.jpg	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-1_patch_192.jpg	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-1_patch_211.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino
Area02_30000-1-1_patch_231.jpg	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-1_patch_232.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste
Area02_30000-1-1_patch_233.jpg	Treino	Treino	Treino	Teste	Treino	Treino	Treino	Treino	Treino	Treino
Area02_30000-1-1_patch_274.jpg	Treino	Treino	Treino	Treino	Treino	Treino	Treino	Teste	Treino	Treino