

Universidade de Brasília – UnB
Faculdade UnB Gama – FGA
Engenharia Eletrônica

**Uso de arquitetura embarcada para automação
do processo de aquisição de imagens para
escaneamento 3D de baixo custo**

Autor: Vitor Meireles Oliveira
Orientador: Prof. Dr. Diogo Caetano Garcia

Brasília, DF
2018



Vitor Meireles Oliveira

**Uso de arquitetura embarcada para automação do
processo de aquisição de imagens para escaneamento 3D
de baixo custo**

Monografia submetida ao curso de graduação
em Engenharia Eletrônica da Universidade
de Brasília, como requisito parcial para ob-
tenção do Título de Bacharel em Engenharia
Eletrônica.

Universidade de Brasília – UnB

Faculdade UnB Gama – FGA

Orientador: Prof. Dr. Diogo Caetano Garcia

Brasília, DF

2018

Vitor Meireles Oliveira

Uso de arquitetura embarcada para automação do processo de aquisição de imagens para escaneamento 3D de baixo custo/ Vitor Meireles Oliveira. – Brasília, DF, 2018-

58 p. : il. (algumas color.) ; 30 cm.

Orientador: Prof. Dr. Diogo Caetano Garcia

Trabalho de Conclusão de Curso – Universidade de Brasília – UnB
Faculdade UnB Gama – FGA , 2018.

1. Automação. 2. Escaneamento 3D. I. Prof. Dr. Diogo Caetano Garcia. II. Universidade de Brasília. III. Faculdade UnB Gama. IV. Uso de arquitetura embarcada para automação do processo de aquisição de imagens para escaneamento 3D de baixo custo

CDU 02:141:005.6

Vitor Meireles Oliveira

Uso de arquitetura embarcada para automação do processo de aquisição de imagens para escaneamento 3D de baixo custo

Monografia submetida ao curso de graduação em Engenharia Eletrônica da Universidade de Brasília, como requisito parcial para obtenção do Título de Bacharel em Engenharia Eletrônica.

Trabalho aprovado. Brasília, DF, 11 de dezembro de 2018 – Data da aprovação do trabalho:

Prof. Dr. Diogo Caetano Garcia
Orientador

**Prof. Dr. Marcus Vinícius Chaffim
Costa**
Convidado 1

Prof. Dr. Renan Utida Ferreira
Convidado 2

Brasília, DF
2018

Este trabalho é dedicado à minha família.

Agradecimentos

Agradeço a Deus por toda a jornada já vivida, sei que sem a confiança e paz dada por Sua presença, jamais teria chegado até aqui.

Aos meu pais, Antônio e Edinalva, por terem feito tudo para que eu me tornasse, profissionalmente, o que vocês sonharam e não podendo para si, investiram em mim, e principalmente por terem me ensinado qualidades de caráter que são herança eterna.

Agradeço ao meu irmão Higor, que de forma única, conseguiu me alegrar e motivar em diversas situações.

Agradecimentos por Deus ter me dado a graça de conhecer minha mui preciosa e amada Joana, me trouxe amadurecimento, motivação e inspiração na hora certa.

Agradeço aos meus estimados amigos, professores e ao meu orientador, mais que importantes para a existência desse trabalho.

*“Mas o nobre projeta coisas nobres
e na sua nobreza perseverará.”
(Bíblia Sagrada, Isaías 32, 8)*

Resumo

O conceito de imageamento 3D tem se popularizado atualmente e passado para um âmbito cada vez mais próximo dos consumidores. Os sensores de profundidade, que são os principais elementos do processo, são facilmente encontrados, de forma direta ou indireta, pois até mesmo video-games fazem uso dessa tecnologia. Assim, tal processo de popularização não se deu apenas pelo crescente uso de impressoras 3D como também pelo fato de grandes empresas terem introduzido no mercado a definição e funcionalidade da realidade aumentada. A construção da imagem-objeto 3D ocorre de diversas maneiras, uma das formas é fazer a leitura do objeto por imagem ou vídeo a partir de diversas posições e obter *frames* para construção da imagem 3D em computador. Todavia, esse processo executado de forma manual é impertinente, o que torna relevante a introdução de automatização, tendo em vista que uma tecnologia será aplicada sobre o processo de escaneamento sem assistência humana manual. O objetivo desse trabalho é entender o funcionamento do processo de escaneamento 3D e propor a automatização da obtenção e envio de imagens, evitando movimentações manuais pelo o usuário e permitindo que nuvem de pontos sejam geradas com o processo de formação do objeto 3D, feito em um computador pessoal através do uso de *softwares* com a técnica de fotogrametria. Além do motor para movimento, um *Raspberry Pi* é usado para fazer a aquisição das imagens e para mostrar uma interface de controle com *touchscreen* para o usuário. Baseado em um *system-on-chip*, a ferramenta proposta vai conter uma plataforma giratória para o posicionamento de objetos, além da câmera, oferecendo o envio das fotos de forma automatizada para posterior geração de objetos tridimensionais.

Palavras-chave: automação. escaneamento 3D. fotogrametria, nuvem de pontos.

Abstract

The concept of 3D imaging has become popular today and getting closer of consumers. The depth sensors, which are the main elements of the process, are easily found, directly or indirectly, because even video-games make use of this technology. Thus, this process of popularization was not only due to the increasing use of 3D printers but also by the fact that large companies introduced the definition and functionality of augmented reality. The construction of the 3D image-object occurs in several ways, one way is to read the object by image or video from several positions and to obtain frames for the construction of the 3D image on a computer. However, this manually executed process is impertinent, which makes the introduction of automation relevant, since a technology will be applied to the scanning process without manual human assistance. The objective of this work is to understand the operation of the 3D scanning process and propose the automation of obtaining and sending images, avoiding manual movements by the user and allowing cloud of points to be generated with the process of forming the 3D object, all this is going to be made in a personal computer through the use of programs. with the photogrammetry technique. In addition to the motion engine, a Raspberry Pi is used to make the acquisition of the images and to show a control interface with touchscreen for the user. Based on a system-on-chip, the proposed tool will contain a rotating platform for positioning objects, in addition to the camera, offering the sending of the photos in an automated way for later generation of three-dimensional objects.

Key-words: automation. 3D scanning. photogrammetry, point cloud.

Lista de ilustrações

Figura 1 – Técnica de fotogrametria aplicada (3DFLOW, 2018)	28
Figura 2 – Raspberry Pi 3	34
Figura 3 – PiCamera V2	35
Figura 4 – Driver de motor de passo A4988	36
Figura 5 – Motor de passo para automatizar a rotação	37
Figura 6 – Cabo flat utilizado nas comunicações	37
Figura 7 – Display para Raspberry utilizado para a GUI	38
Figura 8 – Uso de interface gráfica com comandos	38
Figura 9 – Fonte de potência para alimentação geral do dispositivo	39
Figura 10 – LED’s automotivos utilizados para focar o objeto a ser escaneado	39
Figura 11 – Duty cycle em PWM (SPARKFUN, 2018).	40
Figura 12 – Transistor IRFZ44N utilizado no PWM	40
Figura 13 – Esquemático do sistema proposto	41
Figura 14 – Diagrama de comunicação do sistema proposto	42
Figura 15 – Diagrama da interface do usuário	43
Figura 16 – Diagrama da comunicação siri para Raspberry	43
Figura 17 – a) Fluxograma geral dos processos. b) Fluxograma de captura de imagens. c) Fluxograma de envio de imagens para nuvem.	44
Figura 18 – Modelo isométrico da estrutura da plataforma de escaneamento	45
Figura 19 – Circuito geral da placa de acionamento	45
Figura 20 – Vista da placa de acionamento	46
Figura 21 – Optoacoplador de segurança, 4N25	46
Figura 22 – Plataforma automatizada de aquisição de imagens	49
Figura 23 – Conexão de Raspberry Pi 3 e 4N25	50
Figura 24 – PCB de acionamento com optacopladores	50
Figura 25 – Interface de usuário desenvolvida	51
Figura 26 – Imagens enviadas para Dropbox	51
Figura 27 – Momento de captura de imagens para gerar o modelo 3D	52
Figura 28 – Representação da posição espacial no momento que as fotos foram obtidas	53
Figura 29 – Modelo reconstruído com filtros do software <i>MeshLab</i>	53
Figura 30 – Nuvem de pontos comprovando o resultado e eficiência da captura das imagens do modelo 3D. a) Modelo original. b) Modelo em nuvem de pontos gerado. c) Modelo em nuvem de pontos gerado e tratado	54

Lista de abreviaturas e siglas

CAD	Computer Aided Design
CLI	Comand-line interface
CPU	Central Processing Unit
CSI	Camera Serial Interface
DSI	Display Serial Interface
EPROM	Erasable Programmable Read-Only Memory
FPGA	Field-programmable gate array
GPU	Graphics Processing Unit
GUI	Graphical User Interface
IoT	Internet of Things
IR	Infrared
LAN	Local Architecture Network
LED	Light Emitting Diode
MDF	Medium Density Fiberboard
MVS	Multi-View Stereo
RAM	Random Access Memory
RGB	Red, Green and Blue
RP	Raspberry Pi
PCB	Printed Circuit Board
PWM	Pulse Width Modulation
SLR	Single Lens Reflex
SoC	System on Chip
USB	Universal Serial Bus

Lista de símbolos

Hz	Hertz, unidade de frequência
V	Volts, unidade de tensão elétrica
A	Ampere, unidade de corrente elétrica
g	gramas, unidade de massa
mW	miliWatt, medida de potência

Sumário

1	INTRODUÇÃO	23
1.1	Contextualização	23
1.2	Descrição do problema	23
1.3	Justificativa	24
1.4	Objetivos	24
1.4.1	Objetivo geral	24
1.4.2	Objetivos específicos	24
2	FUNDAMENTAÇÃO TEÓRICA	25
2.1	Nuvem de pontos	25
2.2	Escâner 3D	25
2.2.1	Modelos e formas de escaneamento	26
2.2.1.1	Escâner por triangulação	26
2.2.1.2	Escâner manual com LED de infravermelho	26
2.2.1.3	Escâner de luz estruturada	26
2.3	Fotogrametria	27
2.4	Sistemas embarcados	28
2.4.1	Arquiteturas de sistemas embarcadas	29
2.4.1.1	Microcontrolador	29
2.4.1.2	FPGA	29
2.4.1.3	System-on-chip	30
2.5	Softwares para fotogrametria	30
2.5.1	VisualSFM	31
2.5.2	Meshroom	31
2.5.3	Colmap	31
2.5.4	MeshLab	31
2.5.5	Blender	32
3	DESENVOLVIMENTO	33
3.1	Métodos e tecnologias	33
3.1.1	Interface	33
3.1.2	Estrutura	33
3.2	Componentes e Arquitetura da solução implementada	34
3.2.1	RaspberryPi 3 Model B+	34
3.2.2	Pi Camera V2	35
3.2.3	Driver A4988	36

3.2.4	Motor de passo	36
3.2.5	Fios e cabos	37
3.2.6	Display RaspberryPi	37
3.2.7	Interface de usuário	38
3.2.8	Fonte de potência	38
3.2.9	LED's	39
3.2.10	PWM	39
3.2.11	Armazenamento em nuvem	40
3.2.12	Softwares de tratamento da nuvem de pontos	41
3.3	Solução proposta	41
3.3.1	Arquitetura do sistema	42
3.3.2	Dropbox	42
3.3.3	Interface gráfica do usuário	42
3.3.4	Raspberry	43
3.4	Implementações	44
3.4.1	Estrutura	44
3.4.2	Circuito	45
3.4.3	Interface gráfica para o usuário	46
3.4.4	Pinagem Raspberry Pi 3	48
4	RESULTADOS E CONCLUSÕES	49
4.1	Resultados obtidos	49
4.1.1	Estrutura	49
4.1.2	Placa para acionamento de driver	49
4.1.3	Interface	50
4.1.4	Envio para Dropbox	51
4.1.5	Eficiência das imagens demonstrada em nuvem de pontos	52
5	CONCLUSÕES	55
	REFERÊNCIAS	57

1 Introdução

1.1 Contextualização

Atualmente, o conceito de automatização tem se tornado cada vez mais presente em todo o globo por facilitar processos e trazer maior autonomia à vida das pessoas. A implementação da automatização pode ser entendida como uma aplicação de tecnologia sobre uma ação de forma que gere uma diminuição ou extinção da assistência humana. Dentre os benefícios da automatização estão: economia de mão-de-obra, economia de custos de eletricidade, economia de custos de materiais, e melhorias na qualidade, precisão e acurácia. Nesse contexto, inclui-se a ideia de automatizar um processo de alta usabilidade, por conta de seu conceito, necessidade e aplicações: o escaneamento de objetos tridimensionais (SITNIK MACIEJ KARASZEWSKI; RUTKIEWICZ, 2010).

Os escâneres 3D têm sido utilizados na indústria há décadas para engenharia reversa, para estudar a geometria e o uso de objetos existentes. Atualmente, eles são utilizados na arquitetura e no planejamento urbano para melhorar as cidades e preservar o patrimônio cultural. Cientistas usam essa tecnologia para criar um arquivo virtual de objetos de valor e fazer reproduções para serem usadas ao estudar certos artefatos. Em medicina, as varreduras em 3D permitem salvar vidas e aumentar a qualidade criando próteses personalizadas e até implantes (OLIVEIRA, 2017).

Existem escâneres 3D em todas as faixas de preços com um intervalo de qualidade correspondente; em geral, quanto maior o valor, mais fiel é a reprodução. Existem escâneres de vários tipos, desde os que usam câmeras simples até os que usam laser para gerar modelos 3D (BASRI; JACOS, 2003). Para alcançar o objetivo de representar o objeto com considerável fidelidade, são oferecidas diferentes imagens de acordo com diferentes pontos de vista.

1.2 Descrição do problema

Existem vários tipos de escâneres que utilizam diversas técnicas diferentes e cada um possui limitações. O custo de um escâner costuma ser alto e o objeto escaneado fica limitado a proposta oferecida pelo escâner comercial, ou seja, o modelo gerado é específico do escâner utilizado. Caso queira fazer o uso de *softwares* que geram o modelo 3D, o processo não será, a princípio, de forma autônoma.

1.3 Justificativa

Tendo em vista as dificuldades de custo, falta de liberdade quanto a execução do processo e principalmente da dificuldade para obtenção de imagens para realização do escaneamento, o trabalho de automatização do processo e envio de imagens para escaneamento através da técnica de fotogrametria se faz justo.

1.4 Objetivos

1.4.1 Objetivo geral

Este trabalho tem por objetivo realizar a automação do processo de escaneamento 3D com uso de um computador de placa Raspberry Pi e sua câmera, PiCamera, além de enviar as imagens para a nuvem do usuário para criar o modelo 3D com *software* específico.

1.4.2 Objetivos específicos

- Construção de plataforma para colocar objetos a serem escaneados.
- Controlar o giro de plataforma para aquisição de dados de imagem.
- Desenvolver código de aquisição de imagem para o Raspberry Pi.
- Construir interface amigável ao usuário em *display* para controle do processo de automatização.
- Desenvolver código para envio de imagens para Dropbox.

2 Fundamentação Teórica

Neste capítulo está um levantamento do estado da arte e da literatura, relevantes ao trabalho, também estão representados os resultados desta fase do projeto que constitui a junção de informações pertinentes para o desenvolvimento do trabalho. Encontram-se analisados como ocorre atualmente o processo de escaneamento 3D, as tecnologias necessárias para aquisição de dados para uma construção de um modelo 3D, desenvolvimento de interface gráfica ao usuário, motor de passo para a aplicação e *softwares* que transformam as imagens recebidas em um modelo 3D.

2.1 Nuvem de pontos

A nuvem de pontos é uma representação tridimensional muito utilizado para representar objetos e espaços. Os pontos são expressos em um mesmo sistema de coordenadas e formam a superfície de objetos (AGARWAL LARS ARGE, 2018).

Embora as nuvens de pontos possam ser renderizadas e inspecionadas diretamente, as nuvens de pontos geralmente são convertidas em modelos de malha de polígono ou malha triangular, modelos de superfície ou modelos CAD por meio de um processo comumente chamado de reconstrução de superfície (AGARWAL LARS ARGE, 2018).

2.2 Escâner 3D

O escâner 3D é um dispositivo que faz o processo de análise de objetos do mundo real ou também de ambientes e faz a coleta de dados relevantes como o formato, silhueta e, quando possível, a cor do objeto a ser escaneado. Existem diferentes tecnologias utilizadas para o escaneamento 3D, cada tecnologia dispõe de vantagens e desvantagens que podem ser, por exemplo, custo, facilidade de uso e qualidade do modelo 3D gerado (MAKE, 2015). Os dados coletados pelo escâner 3D são utilizados em várias aplicações como engenharia reversa, digitalização de artefatos e também criação de próteses (SITNIK MACIEJ KARASZEWSKI; RUTKIEWICZ, 2010).

O modelo 3D gerado pelo escâner consiste em uma nuvem de pontos sobre a superfície do objeto, trazendo uma espécie de *frame* com o formato do objeto, processo chamado de reconstrução. O escâner 3D coleta informações sobre a distância do objeto que está em seu campo de visão, assim a imagem gerada pelo dispositivo descreve a distância até cada ponto da imagem. Tendo isso como base, infere-se que é necessário tirar várias fotografias para obtenção de imagens dos mais variados ângulos e assim o modelo 3D

poderá ser gerado com maior fidelidade. Surge então uma necessidade de todas as fotos estarem ligadas por um referencial, esse processo é chamado de registro e é feito por *software*, podendo até mesmo ser externo ao equipamento que faz o escaneamento das imagens em si, como é o caso da técnica de fotogrametria ([RUSHMEIER, 2002](#)).

2.2.1 Modelos e formas de escaneamento

Existem vários modelos de escâneres 3D, geralmente divididos entre os manuais e o automáticos. A seguir, os mais comuns.

2.2.1.1 Escâner por triangulação

Este tipo de escâner faz uso de laser infravermelho (IR) para aquisição dos dados e então obtém medidas tridimensionais. Neste modelo, a luz infravermelha é projetada sobre o objeto e uma câmera digital que está localizada a alguns centímetros do emissor laser observa a reflexão no campo de visão. A razão para isso é que caso o ângulo e a posição do laser em relação a câmera, sejam mensurados, é possível medir com precisão a posição do objeto, ponto a ponto. A técnica recebe esse nome por conta do triângulo de informação formado pelo laser, objeto e a câmera. Geralmente esta técnica é automatizada por uma plataforma que gira em torno de um eixo ([MAKE, 2015](#)).

2.2.1.2 Escâner manual com LED de infravermelho

Esse tipo de escâner é bastante semelhante ao por triangulação, a diferença está na manipulação manual durante a aquisição 3D. Esse modelo usa um LED infravermelho para analisar a distância entre o objeto e o escâner e assim construir a imagem 3D. O resultado é a chamada nuvem de pontos que em seguida é transformada por software na chamada malha. Se esse escâner incluir uma câmera RGB, efeitos de textura e coloração podem ser adicionados a imagem tridimensional. Nesse modelo a luz ambiente tem grande influência na resolução da imagem e afeta a acurácia das cores ([MAKE, 2015](#)).

2.2.1.3 Escâner de luz estruturada

O modelo por luz estruturada é extremamente preciso e rápido. Nesse modelo, um vídeo projetor "varre" a superfície do objeto criando padrões ou grades. Então, a sequência projetada é observada por uma câmera e a deformação da superfície é calculada. A área do objeto, a depender do modelo, pode ser escaneada em menos de 1 segundo. Alguns modelos possuem uma câmera RGB adicional para acrescentar informações sobre a cor do objeto ([MAKE, 2015](#)).

2.3 Fotogrametria

Fotogrametria é a técnica utilizada neste trabalho e consiste na medição por meio do uso de imagens capturadas. Em um escâner essa coleta de imagens é muito importante, pois infere-se as medidas, distâncias e posições dos pontos na superfície, na nuvem de pontos.

Segundo a ASPRS do inglês *American Society for Photogrammetry and Remote Sensing*, fotogrametria é uma arte, tecnologia e ciência de obtenção de informação confiável de objetos e ambientes através de processo de obtenção contínua de imagens, medindo e interpretando as imagens e padrões. A técnica faz uso das coordenadas 3D para definir a posição dos pontos do objeto 3D no espaço. A orientação externa da câmera define sua localização no espaço e sua direção de visualização. A orientação interna define os parâmetros geométricos do processamento da imagem. Estas orientações são necessárias para o foco da lente da câmera ([MACHINE, 2018](#)).

As soluções de aquisição por fotogrametria 3D não são parte da família de digitalização 3D, como já dito, fotogrametria se traduz em uma técnica. O princípio é capturar várias imagens de um objeto de todos os ângulos e visualizações acessíveis. As imagens são então enviadas para um software usando as cores, sombras e texturas como referência para reconectá-las a um volume tridimensional. É o princípio da estereoscopia (como na visão humana) que é usado para calcular a distância entre cada visualização ([MACHINE, 2018](#)). Existem sistemas que podem acionar várias câmeras de uma só vez para obter um resultado mais preciso e espontâneo. Esses sistemas geralmente usam câmeras SLR do inglês *Single Lens Reflex*, mas alguns protótipos usam câmeras simples de celulares organizadas regularmente em uma estrutura esférica ou cilíndrica. Fotogrametria é a técnica que permite a renderização mais exata das cores.

A fotogrametria é usada em campos como mapeamento topográfico, arquitetura, engenharia, manufatura, controle de qualidade, investigação policial, patrimônio cultural e geologia. Os arqueólogos fazem uso da técnica para produzir rapidamente planos de locais grandes ou complexos, e os meteorologistas usam para determinar a velocidade do vento nos tornados quando os dados meteorológicos objetivos não podem ser obtidos. A fotogrametria também é comumente empregada na engenharia de colisões, especialmente com automóveis. Quando o litígio por acidentes ocorre e os engenheiros precisam determinar a deformação exata presente no veículo, é comum que vários anos se tenham passado e a única evidência que resta são fotografias de cenas de acidentes tomadas pela polícia. A fotogrametria é usada para determinar quanto o carro em questão foi deformado, o que se relaciona com a quantidade de energia necessária para produzir essa deformação. A energia pode então ser usada para determinar informações importantes sobre a colisão (como a velocidade no momento do impacto) ([SUŽIEDELYTÈ-VISOCKIENÈ; JÜRATÈ; BAGDŽIŪNAITÈ, 2015](#)).

A figura 1 demonstra, à esquerda, tetraedros representando a posição da câmera no momento de captura das imagens e um carro formado por uma nuvem de pontos, como objeto ao centro. À direita está representado no que consiste a técnica, com a câmera capturando *frames* em volta do objeto.

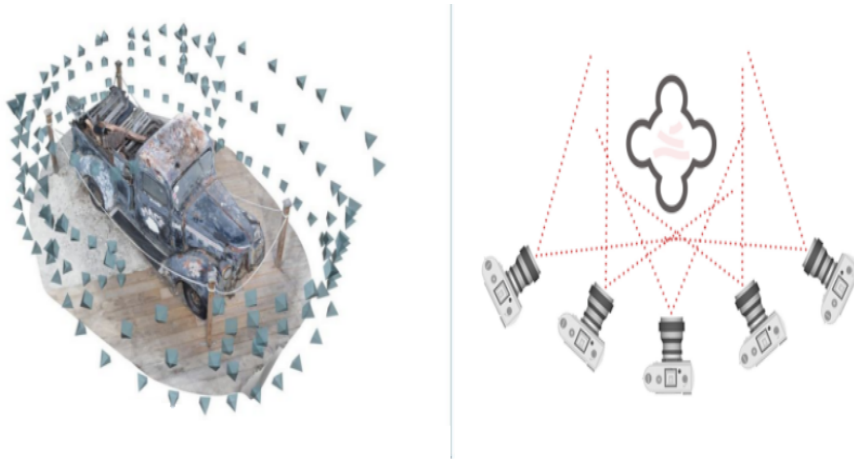


Figura 1 – Técnica de fotogrametria aplicada (3DFLOW, 2018)

2.4 Sistemas embarcados

Os sistemas embarcados estão por toda parte, eles revolucionam continuamente diversas áreas do conhecimento humano e eles estão presentes nos mais diversos dispositivos que envolvem eletrônica.

Os sistemas embarcados podem ser entendidos como um computador construído com um propósito em sua aplicação, ao invés de prover um sistema computacional generalizado. Essa definição independe da sua construção e pode abranger desde lógicas combinacionais até super processadores multi-cores (EMBARCADOS, 2018).

O termo tem sua exclusividade, não caracterizando o sistema embarcado como algo de eletrônica geral, sua nomenclatura se remete à complexidade de desenvolvimento e os poderes computacionais que diferenciam um sistema embarcado de um sistema sem habilidade computacional.

O primeiro sistema embarcado de grande conhecimento foi o AGC do inglês *Apollo Guidance Computer*, ele era o controlador geral das espaçonaves Apollo, mas este ainda não possuía processador, pois era todo feito com portas NOR e operava a 1024MHz. Desde aquela época os sistemas embarcados evoluíram bastante e atualmente tem muito mais poder de processamento e atinge uma maior gama de aplicações. Dentre as mais diversas aplicações, têm-se a indústria automotiva, médica, comercial e militar (EMBARCADOS, 2018).

Além da característica de serem projetados para desempenharem uma tarefa específica, os sistemas embarcados também podem, de maneira mais rara, serem usados para multi tarefas. Alguns possuem desempenho considerado em tempo real, outros já possuem restrição a isso. Vários sistemas embarcados também podem se unir para gerar uma solução de propósito geral, algo muito comum em carros mais modernos e tecnológicos.

Um conceito relevante quanto aos sistemas embarcados são as instruções de programação escritas para o sistema, elas são referidas como *firmware* e são armazenadas em memória do tipo *read-only* ou *flash*. Essas instruções são executadas em um hardware muito limitado e são importantíssimas para esses sistemas.

2.4.1 Arquiteturas de sistemas embarcadas

A seguir as principais arquiteturas de sistemas embarcados.

2.4.1.1 Microcontrolador

Os microcontroladores são uma parcela importante dos sistemas embarcados, eles possuem interface de entrada e saída com dispositivos elétricos variados, como relés, transistores e outros. Microcontroladores se diferem de microprocessadores de uso geral, pois geralmente são dedicados a projetos de computadores com periféricos de entrada e saída como teclado, mouse e monitor. As aplicações de microcontroladores no início eram mais incipientes e voltadas para a indústria, mas com o tempo se propagaram para as mais diversas áreas de atividade humana (OLIVEIRA, 2017).

As funcionalidades agregadas a esses dispositivos inicialmente se resumiam a interfaces de entrada e saída (I/O) e aos poucos foram agregando, a cada nova versão ou novo produto, mais memória RAM, mais memória EPROM para programas e dados e circuitos de oscilador(clock), interfaces de comunicação mais variadas, como serial e USB e, mais recentemente, interfaces de rede, como Ethernet, WiFi e Bluetooth.

Um microcontrolador com essas várias funcionalidades se torna capaz se executar diversas aplicações. Assim a questão energética se torna outro fator a ser pensado, apesar de precisar de pouca energia, algumas vezes há a necessidade do consumo ser tão baixo ao ponto de não ter manutenção necessária para por longo tempo, sem nenhuma intervenção, mas sempre executando na hora certa. O custo desses dispositivos varia com diversos fatores, mas entre os mais comuns estão a tecnologia empregada, a capacidade de controle e a capacidade de memória e processamento (OLIVEIRA, 2017).

2.4.1.2 FPGA

FPGA do inglês *field-programmable gate array*, é um circuito integrado projetado para ser programado após a sua manufatura. Geralmente a programação de uma FPGA

se dá por uma linguagem de descrição de *hardware*.

Esses dispositivos são usados para solucionar diversos problemas da computação. Algumas soluções são facilmente providas pela FPGA, pela implementação de um pequeno processador com a vantagem de ser incrivelmente rápido para algumas aplicações, isso ocorre por conta da natureza paralela e otimizada pelas várias portas da FPGA usadas em certos processos (TORONTO, 2018).

Uma grande aplicação das FPGA's atualmente é com algoritmos de aprendizado de máquina e redes neurais. O processo paralelo de aprendizagem e tentativa, ainda que em uma frequência de operação relativamente menor, redunde em grande eficiência se comparado com outros tipos de sistemas embarcados (MITTAL, 2018).

2.4.1.3 *System-on-chip*

Esse sistema embarcado é um circuito integrado em chip que integra todos os componentes de um computador ou outro sistema eletrônico. Esses dispositivos, geralmente, são compostos de uma central de processamento(CPU), memória, interfaces de entrada e saída de dados e sinais e, algumas vezes, memória de armazenamento. Pelo fato de todos esses aparatos estarem integrados em uma única placa, os SoCs costumam consumir bem menos energia que outros dispositivos que possuem a mesma aplicação em vários *chips* (EETIMES, 2018).

Dentre diversas qualidades, a vantagem do baixo consumo de energia faz com que esses dispositivos tenham vasta aplicação em computação envolvendo dispositivos móveis, sistemas embarcados gerais e também na criação de tecnologias envolvendo IoT do inglês *Internet of Things* (REUSE, 2018).

Esses sistemas integram um microcontrolador ou microprocessador com vários periféricos e essa é uma razão para sua alta aplicabilidade. O Raspberry Pi é um dos sistemas em placa mais populares e será utilizado nesse trabalho, pois possui vários periféricos integrados que são necessários ao projeto.

2.5 Softwares para fotogrametria

Esta seção fala sobre alguns software que podem ser usados para gerar o modelo 3D. Serão destacados softwares gratuitos.

O funcionamento dos *softwares* é baseado nas técnicas de fotogrametria. Comparações são feitas com todas as fotos e o objeto é tomado como o centro, então o modelo 3D é calculado.

2.5.1 VisualSFM

O *Visual Structure From Motion* é o programa GUI para reconstrução 3D usando a técnica SFM do inglês *Structure From Motion*. O sistema de reconstrução integra vários projetos como uso de GPU e ajuste para uso em vários núcleos. O VisualSFM é executado rapidamente, explorando o paralelismo com vários núcleos para detecção de recursos, correspondência de recursos e ajuste de pacotes (WU, 2018).

Para uma reconstrução densa, este programa integra a execução da cadeia de ferramentas desenvolvidas na Universidade de Washington. A saída SFM do VisualSFM funciona com várias ferramentas adicionais como o MeshLab, podendo ser uma malha ou nuvem de pontos.

2.5.2 Meshroom

O Meshroom é um Software de reconstrução 3D gratuito e de código aberto baseado em um *framework* chamado AliceVision, portanto o ponto-chave do programa é o trabalho do consistente *framework*.

O AliceVision é um framework de visão computacional de fotogrametria que fornece algoritmos de Reconstrução 3D e Rastreamento de Câmera. O AliceVision tem como objetivo fornecer uma sólida base de software com algoritmos de visão computacional de última geração que podem ser testados, analisados e reutilizados. O projeto é resultado da colaboração entre a academia e a indústria para fornecer algoritmos de ponta com a robustez e a qualidade necessárias para o uso da produção (ALICEVISION, 2018). O que difere o *software* dos demais é o utilização programável do uso dos recursos computacionais, além de ser gratuito.

2.5.3 Colmap

O COLMAP é um pipeline Estrutura a Movimento (SfM) e Multi-View Stereo (MVS) de uso geral com uma interface gráfica e de linha de comando. Ele oferece uma ampla gama de recursos para a reconstrução de coleções de imagens ordenadas e não ordenadas. O software está licenciado sob a nova licença BSD (COLMAP, 2018).

2.5.4 MeshLab

Outro software importante é o MeshLab para depois que o modelo tiver sido obtido, haver melhoras como limpeza e aprimoramentos.

O MeshLab é um sistema de código aberto para processamento e edição de malhas triangulares 3D. Ele fornece um conjunto de ferramentas para edição, limpeza, cura, inspeção, renderização, texturização e conversão de malhas. Ele oferece recursos para

processar dados brutos produzidos por ferramentas ou dispositivos de digitalização 3D e para preparar modelos para impressão 3D ([MESHLAB, 2018](#)).

2.5.5 Blender

Um dos mais conceituados *softwares* de renderização e edição 3D é, sem dúvidas, o Blender.

O Blender é o pacote de criação 3D gratuito e de código aberto. Suporta a totalidade do pipeline 3D - modelagem, manipulação, animação, simulação, renderização, composição e rastreamento de movimento, até mesmo edição de vídeo e criação de jogos ([BLENDER, 2018](#)). Por ser tão completo ele se faz útil e é uma excelente opção no caso de pós aquisição do modelo para que o objeto 3D gerado tenha melhor aparência e qualidade.

3 Desenvolvimento

Para aquisição de dados e formação da imagem por escaneamento 3D, são necessários dispositivos e componentes eletrônicos aplicados de certa maneira, como é mostrado no capítulo subsequente. De forma básica, o computador em placa, Raspberry Pi, ficará responsável por automatizar o processo de rotação da plataforma, adquirir as imagens com a *PiCamera* e envio das fotos para o para a nuvem. O processamento será feito *offline*, não em tempo-real e nem próximo disso, pois o Raspberry Pi, a princípio não possui *hardware* viável para fazer o processamento da imagem. Há pelo menos a proposta de dois *softwares* gratuitos para gerar o modelo 3D. Além disso, no capítulo são apresentados detalhes das implementações realizadas levando em consideração aspectos como materiais utilizados, arquiteturas, plataformas de desenvolvimento e ferramentas.

3.1 Métodos e tecnologias

3.1.1 Interface

Para que aplicações se tornem mais fáceis de utilizar e com uma apresentação mais amigável ao usuário, interfaces GUI do inglês *Graphical user interface* são projetadas. A interface gráfica do usuário possui um formato que permite aos usuários interagir com dispositivos eletrônicos por meio de ícones gráficos e indicadores visuais, como notação secundária, em vez de interfaces de usuário baseadas em texto, etiquetas de comando digitadas ou navegação de texto. GUIs foram introduzidas em reação à curva de aprendizado íngreme percebida de interfaces de linha de comando (CLI), que requerem comandos para serem digitados em um teclado de computador, ao invés de *displays* com ícones, por exemplo (HOPE, 2018).

Em uma interface gráfica de usuário as ações são tomadas através dos elementos visíveis. Além de computadores, as GUIs estão presentes em vários dispositivos, dentre os quais sistemas embarcados. Assim, para uma melhor operação do usuário, a GUI se faz muito útil no projeto de automatização do processo de escaneamento 3D, além da atual facilidade devido aos diversos *Frameworks*.

3.1.2 Estrutura

A estrutura da plataforma do escâner é feita de MDF do inglês *Medium Density Fiberboard* para que possa ser transportada sem causar danos e quebras, além de facilitar a aplicação de parafusos para fixação dos dispositivos que fazem parte do sistema automatizado do envio de imagens do escâner. Toda a estrutura é pintada na cor preto fosco com

o objetivo de reduzir interferências por luz e garantir um fundo homogêneo para destaque do objeto a ser escaneado ao centro.

3.2 Componentes e Arquitetura da solução implementada

Nesta seção há descrições sobre componentes e tecnologias empregadas.

3.2.1 RaspberryPi 3 Model B+

O Raspberry Pi (Figura 2) é um computador de placa única com conectividade LAN sem fio e Bluetooth e é utilizado como uma placa de desenvolvimento microcontrolado e microprocessado. O Raspberry Pi 3 é compatível com os padrões de comunicação WI-FI utilizados para envio das fotos que passam pelo processo de fotogrametria, além dessa importante compatibilidade ele possui um específico *display* sensível ao toque e sem ocupação dos pinos, fazendo comunicação por cabo *flat* que facilitará consideravelmente a implementação de ações em interfaces gráficas. A placa possui um conector do tipo USB-C para alimentação em corrente contínua (RASPBERRY, 2018).

A placa tem seu destaque a programação em Python, mas aceita diversas outras como C++ e Java.



Figura 2 – Raspberry Pi 3

As especificações técnicas da placa Raspberry utilizada.

- Broadcom BCM2837B0, Cortex-A53 (ARMv8) 64-bit SoC @ 1.2GHz
- 1GB LPDDR2 SDRAM
- 2.4GHz e 5GHz IEEE 802.11.b/g/n/ac wireless LAN, Bluetooth 4.2, BLE
- GPIO de 40 pinos
- 4 portas USB 2.0

- Porta CSI do inglês *Camera Serial Interface* para se conectar a câmera
- Porta DSI do inglês *Display Serial Interface* para se conectar ao *display touchscreen*
- Porta Micro SD para Sistema operacional e memória para dados
- 5V/2.5A DC como potência de entrada

O sistema utilizado na placa durante todo o trabalho foi a distribuição Linux Raspbian Jessie.

3.2.2 Pi Camera V2

A câmera utilizada para a fotogrametria é a câmera do Raspberry Pi V2 (Figura 3), que provê comunicação perfeita com placa por cabo *flat* sem a necessidade de ocupar pinos. A câmera possui resolução de 8 MegaPixels, o que é suficiente para a aplicação, estimada em resultados obtidos com o estado da arte.

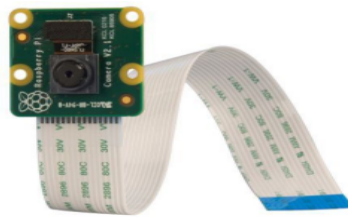


Figura 3 – PiCamera V2

As especificações da câmera utilizada para fotogrametria.

- Tamanho de 25 x 24 x 9 mm
- Resolução de 8 Megapixels
- Sensor Sony IMX219
- Resolução do sensor 3280x2464 Pixels
- Comprimento do foco em 3.04mm
- Campo de visão horizontal em 62,2 graus
- Campo de visão vertical em 48,8 graus
- Taxa de foco em 2.0

3.2.3 Driver A4988

O *driver* do motor de passo escolhido foi o A4988(Figura 4) por ser um driver de motor de micro passos e com conversor já integrado para fácil operação. Ele foi projetado para operar motores bipolares de passo nos modos *full*, *half*, *quarter*, *eighth* e *sixteenth*, com uma capacidade de drive de saída de até 35 V e aproximadamente 2 A, o que é suficiente para a aplicação com o motor utilizado. O A4988 inclui um tempo de inatividade fixo e regulador de corrente que tem a capacidade de operar nos modos de decaimento Lento ou Misto.

O tradutor é a chave para a fácil implementação do A4988. Simplesmente inserir um pulso na entrada STEP aciona o motor em um micro-passo. Não há tabelas de sequência de fase, controles de alta frequência nem programas complexos de interface de rede. A interface A4988 é ideal para aplicações em que um microprocessador complexo não está disponível ou está sobrecarregado, como é o caso da Raspberry executando a interface gráfica, tirando fotos e enviando as imagens para Dropbox (ALLEGRO, 2014).



Figura 4 – Driver de motor de passo A4988

3.2.4 Motor de passo

O motor escolhido para ser usado é o motor de passos síncronos de 6 fios da marca *Tamagawa* (Figura 5). O motor conta com 200 passos e 1.5A. O motor tem força suficiente para rotacionar a plataforma com o objeto de peso estimado em até 500g, segundo testes. Apesar da principal característica do motor de passo ser a precisão, a escolha desse motor para o projeto não visa só a precisão, mas sim um controle mais ajustado acerca da velocidade de rotação, pois isso influenciará diretamente os resultados de escaneamento por se tratar da taxa de captura de imagens.

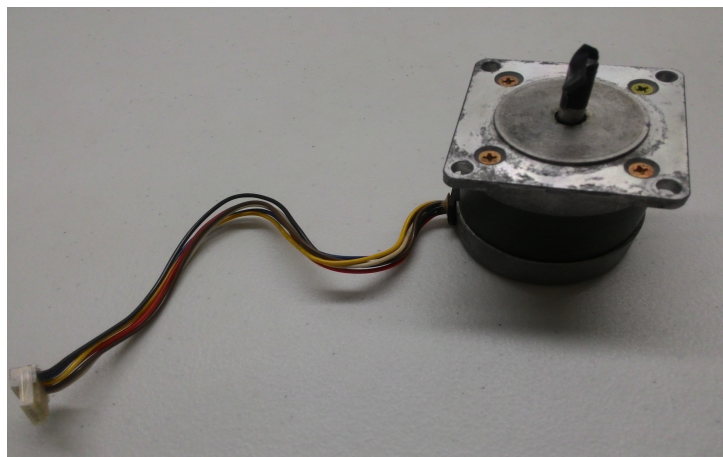


Figura 5 – Motor de passo para automatizar a rotação

3.2.5 Fios e cabos

Para a comunicação entre as placas estão sendo usados cabos *flat* (Figura 6) com valor de 26 na escala AWG na tabela de fios e cabos, com exceção dos fios que acionam as bobinas do motor de passo e os LED's de iluminação que passam maior corrente e precisam um fio tabelado em 22 AWG.

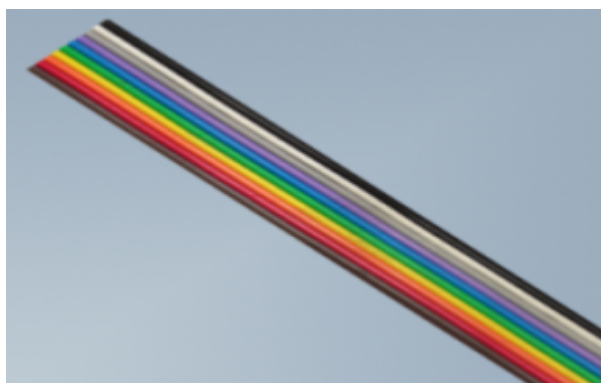


Figura 6 – Cabo flat utilizado nas comunicações

3.2.6 Display RaspberryPi

O *display touchscreen* da Raspberry, apresentado na Figura 7, é uma peça importante do projeto, pelo fato de trazer a parte gráfica ao usuário e seus controles. Pela tela, várias características podem ser expressas. O *display* é de 7 polegadas com tela com resolução de 800x480 *pixels* o que possibilita a criação de um ambiente de excelente interação com o usuário.

O *display* é conectado a uma placa adaptadora, que se comunica com o Raspberry Pi utilizando o conector DSI (*display*), a alimentação (5V) pode ser feita pela própria

GPIO do Raspberry utilizando dois pinos. Tudo se encaixa perfeitamente para um ótimo *layout* em projetos embarcados, entretenimento e IoT.



Figura 7 – Display para Raspberry utilizado para a GUI

- Dimensões: 19,4 x 11 x 2 cm (incluindo espaçadores)
- Área visível da tela: 15,5 x 8,6 cm

3.2.7 Interface de usuário

A interface é criada utilizando um *framework* em C++ escolhido, chamado QT Creator e partes também são feitas utilizando uma integração com a linguagem *python* utilizando as bibliotecas Pyuic5 e Pyside2.

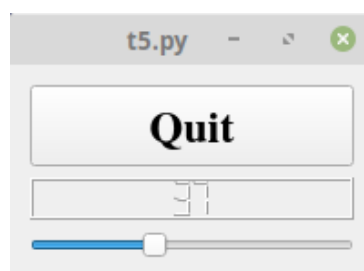


Figura 8 – Uso de interface gráfica com comandos

3.2.8 Fonte de potência

A fonte escolhida para que haja alimentação para o driver fazer o acionamento do motor é de 12v e até 6A (Figura 9). Esta fonte também será conectada para alimentação dos LED's automotivos.



Figura 9 – Fonte de potência para alimentação geral do dispositivo

3.2.9 LED's

Para a centralização do objeto e melhor destaque do objeto nas imagens a serem enviadas serão utilizados LED's automotivos (Figura 10), pois tem alto poder de luminosidade e tem boa relação custo x benefício e trazem bons resultados para a quantidade em caso de controle PWM.



Figura 10 – LED's automotivos utilizados para focar o objeto a ser escaneado

3.2.10 PWM

A técnica por trás do PWM, do inglês *Pulse Width Modulation*, consiste na comunicação de informação através de um pino digital. Nessa técnica uma saída digital produz uma onda quadrada, de forma que a informação é transmitida através da largura do pulso que está em nível alto em um período de tempo. O percentual da largura do pulso em nível alto em relação ao nível baixo do período da onda é chamado *duty cycle*, isso é mostrado na Figura 11.

Dentre as vantagens do PWM pode-se citar a possibilidade de simular um sinal com formato de onda controlado, através de um dispositivo com porta naturalmente digital. Esse tipo de controle será utilizado para deixar a luminosidade dos LED's automotivos de acordo com a necessidade. Isso também pode controlar a velocidade em motores elétricos e outras aplicações. Desse forma um sistema digital pode processar digitalmente as informações e ainda controlar (por meio do PWM) determinado atuador (SPARKFUN, 2018).

Para o PWM dos LED's de iluminação, será utilizado o transistor IRFZ44N (Figura 12), pois ele tem capacidade de suportar a alta potência que os LED's demandam. Ele suporta tensão dreno-fonte(V_{ds}) de 55V, corrente de dreno de 49A e resistência dreno fonte de 17,5m Ohm.

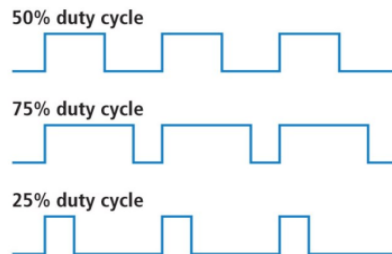


Figura 11 – Duty cycle em PWM (SPARKFUN, 2018).

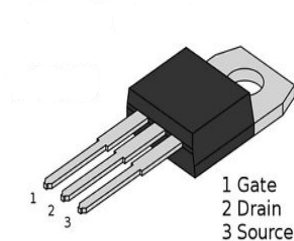


Figura 12 – Transistor IRFZ44N utilizado no PWM

3.2.11 Armazenamento em nuvem

O serviço de armazenamento em nuvem refere-se à utilização dos poderes computacionais de servidores hospedados em *Datacenters* conectados pela internet (GARTNER, 2018). Dentre os serviços mais populares estão o Dropbox, Google Drive e OneDrive. Para a aplicação é utilizado o Dropbox devido a aplicabilidade utilizando bibliotecas em Python 3.

O Dropbox é um serviço de hospedagem de arquivos que oferece armazenamento em nuvem, sincronização de arquivos, nuvem pessoal e software cliente. O Dropbox pode criar uma pasta especial no computador do usuário, cujos conteúdos são então sincronizados para os servidores do Dropbox e para outros computadores e dispositivos nos quais o usuário instalou o Dropbox, mantendo os mesmos arquivos atualizados em todos os dispositivos (DROPBOX, 2018).

O serviço de Dropbox será utilizado para enviar as fotos tiradas pela PiCamera e deixar na nuvem para o usuário fazer o download em um computador e realizar a criação do modelo 3D utilizando software específico.

A biblioteca Dropbox-Uploader em Python 3 é utilizada para fazer o envio.

3.2.12 Softwares de tratamento da nuvem de pontos

Dentre os softwares utilizados para gerar o modelo 3D através da técnica de fotogrametria estão o MeshRoom, que é mais focado em objetos maiores e principalmente o VisualSFM que serve perfeitamente para a criação do modelo, ainda com objetos de tamanho considerados pequeno. Ambos os softwares são gratuitos e podem fazer necessário o uso de um renderizador como o Blender.

3.3 Solução proposta

De forma geral, a solução de automatização é uma plataforma parcialmente fechada para efeitos de melhor controle de luminosidade interna e com uma pequena plataforma em formato circular para efetuar a rotação e obter várias fotos em diversas posições. Com as fotos tiradas, elas devem ser enviadas para o *Dropbox* e então deverá ser feito o tratamento necessário para geração do modelo 3D através de um computador utilizando *software* específico.

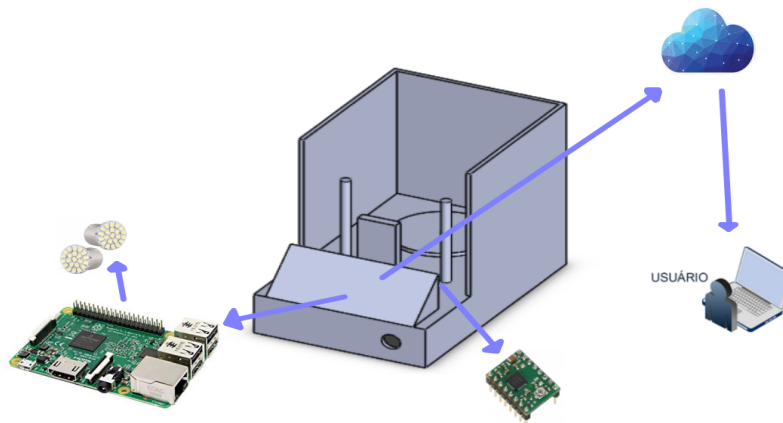


Figura 13 – Esquemático do sistema proposto

A caixa para aquisição automatizada das fotos conta com os seguintes componentes de hardware: Uma Raspberry Pi 3 que faz o controle PWM de iluminação, manda os sinais para o *driver* rodar o motor de passo e ainda adquirir as imagens e as envia. A interface gráfica do usuário é implementada em um computador pessoal e ela faz todo o controle das ações em seus devidos momentos. A comunicação é um fator muito importante no trabalho, tendo em vista que há um dispositivo inicial que é responsável por comunicar seus entes e proporcionar a criação de boas fotos para que a fotogrametria seja feita com sucesso. De outro lado, o recebimento das imagens por Dropbox deve se fazer completamente viável para que o software tenha boas fotos e em boas qualidades. Toda a estrutura física também foi desenvolvida no trabalho, de forma a tornar algo o mais próximo de um protótipo comercial.

3.3.1 Arquitetura do sistema

A Figura 14 demonstra a proposta organizacional do projeto. A justificativa da escolha da Raspberry Pi 3 (representada na Figura 14 por RasPi3) se dá pelo fato dela ter um *display* que torna a criação da interface facilitada e ela possui *hardware* satisfatório para execução das tarefas sem a necessidade de módulos externos como, por exemplo, o envio das imagens por rede.

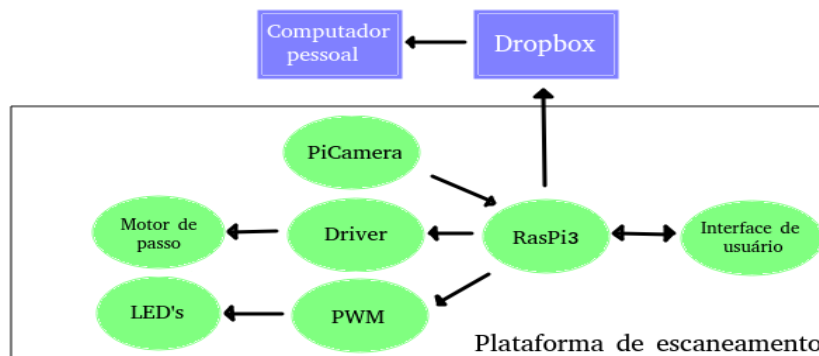


Figura 14 – Diagrama de comunicação do sistema proposto

3.3.2 Dropbox

Para automatização do envio das imagens está sendo utilizado o DropBox Uploader, um BASH script que pode ser usado para enviar, baixar, deletar, listar arquivos e outras funções do Dropbox, ele também sincroniza o que é muito relevante para o trabalho.

O Dropbox Uploader fará o trabalho semelhante ao de um servidor, armazenando as imagens. A comunicação com o Dropbox se dá pela interface criada, nenhum comando é digitado pelo usuário.

3.3.3 Interface gráfica do usuário

A interface gráfica permite ao usuário executar ações por meio de comandos. Através do display sensível ao toque, a operação de capturar as imagens começa e após a retirada de todas as imagens outro botão é responsável pelo envio para Dropbox.

A Figura 15 demonstra a interface que a GUI propicia e o fluxo de informação e comandos.

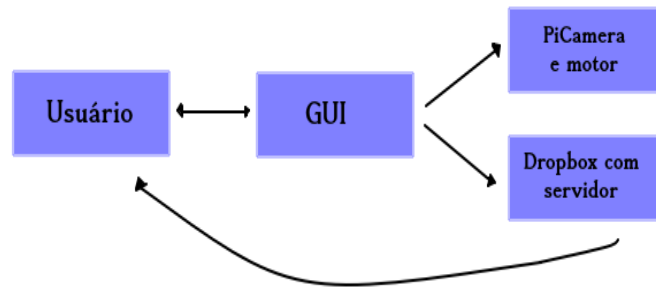


Figura 15 – Diagrama da interface do usuário

3.3.4 Raspberry

O Raspberry Pi 3 é responsável por controlar o driver de potência para o motor de passo(A4988), gerar o sinal PWM e capturar as fotos, além de projetar toda a interface do usuário e fazer o envio de imagens para Dropbox. A Raspberry já será iniciada no programa para controle pelo usuário dos passos do escaneamento.

Na Figura 16 estão caracterizadas as conexões físicas, por fios, e também as construídas virtualmente com o Dropbox e visual no *display*.

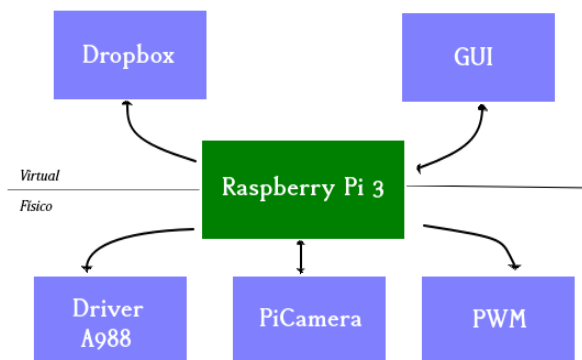


Figura 16 – Diagrama da comunicação siri para Raspberry

No diagrama à esquerda na Figura 17 é apresentado o fluxograma em que estão descritos os passos do algoritmo de execução para a captura das imagens e seu envio para a nuvem.

No começo do processo a primeira ação é capturar várias imagens. Com as imagens no cartão de memória utilizado pela Raspberry Pi, então elas podem ser enviadas para o Dropbox. Após o download das fotos, em um computador pessoal, determinados softwares podem usar a técnica de fotogrametria para que o modelo 3D seja gerado.

Na figura 17 ao centro está caracterizada uma sub-rotina para a captura de imagens. A execução, após o clique no botão pelo usuário, é toda gerenciada pelo Raspberry

Pi. A função dá início com a PiCamera que tira uma foto inicial, então o motor dá um giro de α_r graus, conforme a Equação 3.1, esse processo se repete até que uma volta completa se conclua(360°) e então o processo termina e todas as fotos estão prontas para serem enviadas.

$$K_r = \frac{360}{\alpha_r} \quad (3.1)$$

Para o fluxograma à direita na Figura 17 está exposto o processo de envio das imagens para o Dropbox. As imagens são enviadas pela Raspberry Pi conectado a rede, cada uma por vez. A primeira imagem é enviada e o contador é incrementado baseado no limitador 'Kr' da Equação 3.1, quando todas as imagens são enviadas o processo chega ao fim e a técnica de fotogrametria pode ser aplicada no computador em que as fotos forem baixadas e o software pode fazer o tratamento para que o modelo 3D enfim seja gerado.

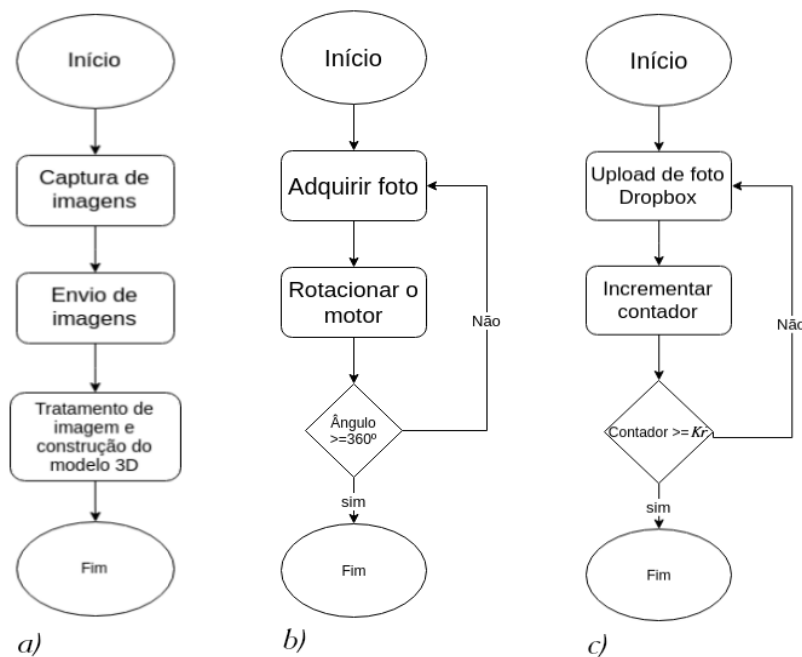


Figura 17 – a) Fluxograma geral dos processos. b) Fluxograma de captura de imagens. c) Fluxograma de envio de imagens para nuvem.

3.4 Implementações

3.4.1 Estrutura

A estrutura da caixa, conforme a Figura 18, foi feita em MDF para proporcionar maior firmeza na movimentação. As partes foram pintadas de preto fosco para evitar problemas com a luz.

O motor e as instalações, como fonte e placas estão na parte de baixo da caixa, equilibrando a altura e melhorando a estética do protótipo. O *display touch* fica na frente para que o usuário possa acessar os comandos. Existem ainda três mini-postes, dois laterais com os LED's de iluminação e um central com a câmera, conforme Figura 18.

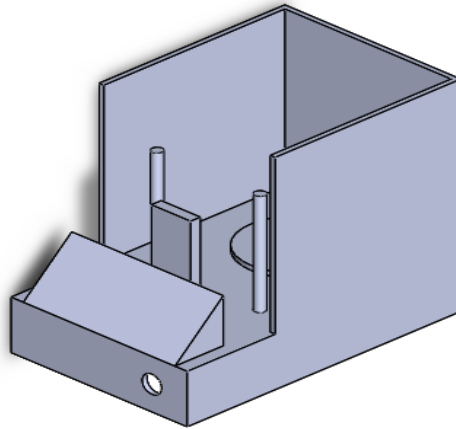


Figura 18 – Modelo isométrico da estrutura da plataforma de escaneamento

3.4.2 Circuito

O circuito da Figura 19 mostra todas as conexões para que ao acionar o *driver* de potência, não tenha nenhuma passagem de corrente indevida para a placa Raspberry que possa danificá-la. A Figura 20 mostra a placa projetada com os componentes e trilhas. É importante destacar o uso principal do circuito integrado 4N25 (Figura 21) como componente de segurança, pois não possui conexão direta de acionamento.

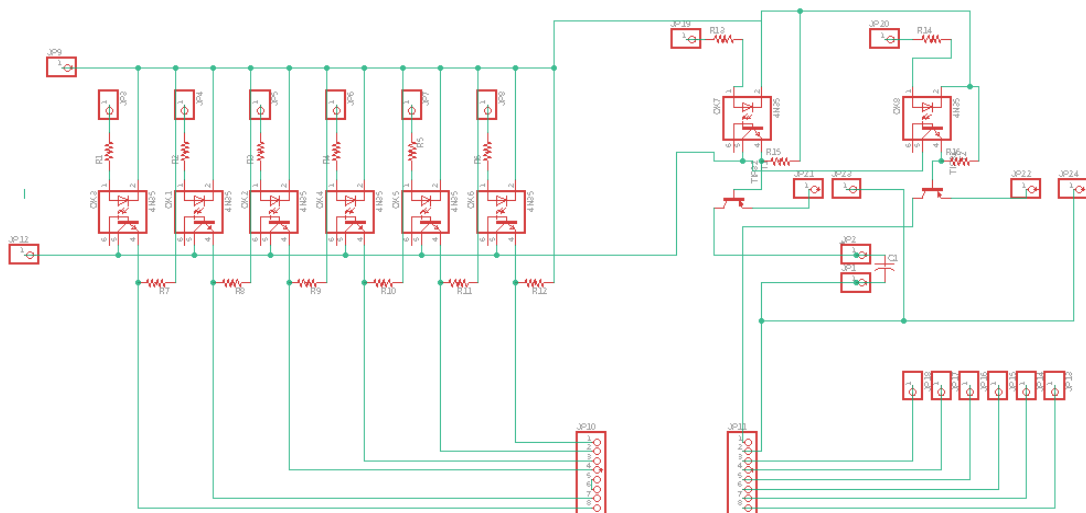


Figura 19 – Circuito geral da placa de acionamento

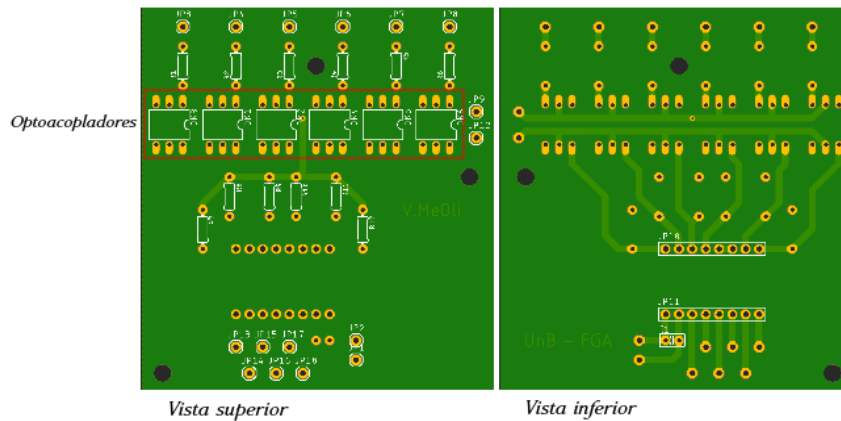


Figura 20 – Vista da placa de acionamento

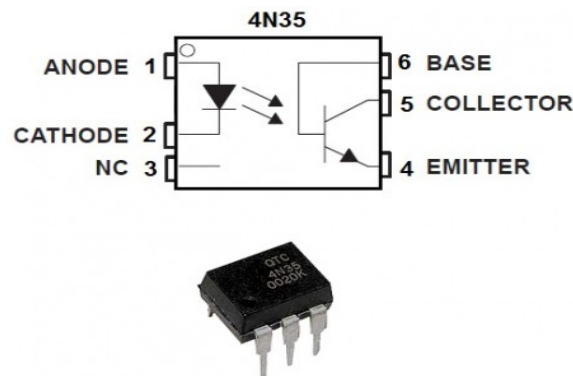


Figura 21 – Optoacoplador de segurança, 4N25

3.4.3 Interface gráfica para o usuário

A interface foi implementada em um notebook da marca CCE com processador core i7 de segunda geração 2670qm e 8Gb de memória RAM, em ambiente Linux Ubuntu e transferida para a Raspberry Pi. As linguagens utilizadas foram C++ e Python no ambiente de desenvolvimento Atom, além do *framework* QT creator.

O programa desenvolvido possui três funções básicas: acionar a captura de imagens, enviar as imagens para Dropbox e também o controle de luminosidade por PWM.

O acionamento do motor se dá conforme o código em Listing 3.1, retratado pela função invocada pelo botão "Tirar fotos" na interface em tela. Nesta mesma função, as fotos são tiradas no momento certo, há um tempo de *delay* para que a câmera estabilize seu foco e faça um balanço com a luminosidade ajustada.

Os códigos utilizados no projeto podem ser encontrados no GitHub: <https://github.com/VitorMeirelesOliveira/scan_tcc>.

Listing 3.1 – Código de captura de imagens

```

#Funcion to take pictures and rotate the step_motor
def pressedTakeButton(self):
    # Just a feedback that the button has just been pressed
    print ("Pressed Take!")
    for n in range(1, 90):
        GPIO.output(ms1, 1)
        GPIO.output(ms2, 1)
        GPIO.output(ms3, 1)

        GPIO.output(enable_r, 0)
        time.sleep(0.0001)
        GPIO.output(dir_r, 0)

        #One only step for take a picture
        for m in range(0, steps_per_rotation_for_motor):
            GPIO.output(step_r, 0)
            time.sleep(0.0005)
            GPIO.output(step_r, 1)
            time.sleep(0.0005)

        GPIO.output(enable_r, 1)

        #Taking a picture
        dirpath = os.getcwd()
        camera.start_preview()
        time.sleep(2)
        camera.capture(dirpath + '/image_%s.jpg' %n)
        camera.stop_preview()
        time.sleep(1)

    # When done, reset Raspberry's pins
    GPIO.cleanup()

```

Após todas as fotos estarem em memória, o botão "Enviar fotos"na interface envia as fotos para o Dropbox conforme o código em Listing 3.2.

Listing 3.2 – Código de envio de imagens para Dropbox

```

#Funcion which send pictures to Dropbox
def pressedTakeButton(self):
    # Just a feedback that the button has just been pressed
    print ("Pressed Take!")
    time.sleep(2)
    os.system('~/.Dropbox-Uploader/dropbox_uploader.sh upload *.jpg /Apps/PythonUp

```

O ajuste de luminosidade por PWM dos LED's pode ser feito através do *slider*

horizontal a qualquer momento, todavia a mudança na iluminação após algumas fotos já terem sido tiradas pode colocar as fotos iniciais e conseqüentemente todas as imagens perdidas, pois a diferença de luz pode atrapalhar o *software* que vai gerar o modelo 3D.

3.4.4 Pinagem Raspberry Pi 3

A seguir as conexões realizadas no Raspberry para que a comunicação com o driver fosse possível. Quanto a comunicação com a PiCamera V2, ela foi feita por cabo *flat*.

Raspberry	A4988
Pino 26	DIR
Pino 19	STEP
Pino 13	ENABLE
Pino 17	MS1
Pino 27	MS2
Pino 22	MS3

4 Resultados e conclusões

Neste capítulo são apresentados os resultados obtidos ao longo do trabalho e os objetivos alcançados com a solução proposta.

4.1 Resultados obtidos

4.1.1 Estrutura

A estrutura foi usinada em MDF e seguiu os planos em desenho CAD inicial. A plataforma giratória ficou firme sobre o motor de passo e apresenta estabilidade quanto ao giro com objeto em cima.

A imagem abaixo apresenta o estado do protótipo.

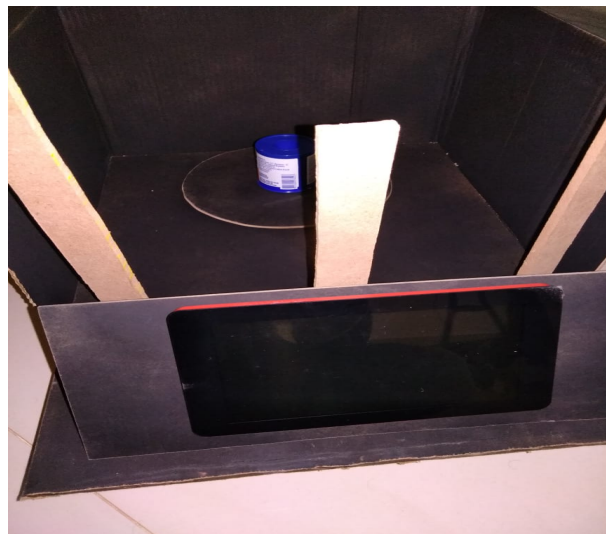


Figura 22 – Plataforma automatizada de aquisição de imagens

4.1.2 Placa para acionamento de driver

A partir do projeto do circuito, a seguinte placa foi feita e funciona com a corrente do Raspberry Pi através dos optoacopladores. Seguindo o modelo da Figura 23 foi criada a placa da Figura 24 para conexão entre Raspberry e o 4N25

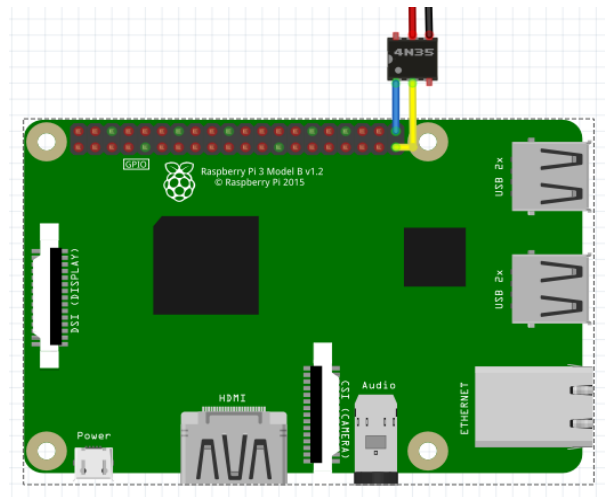


Figura 23 – Conexão de Raspberry Pi 3 e 4N25

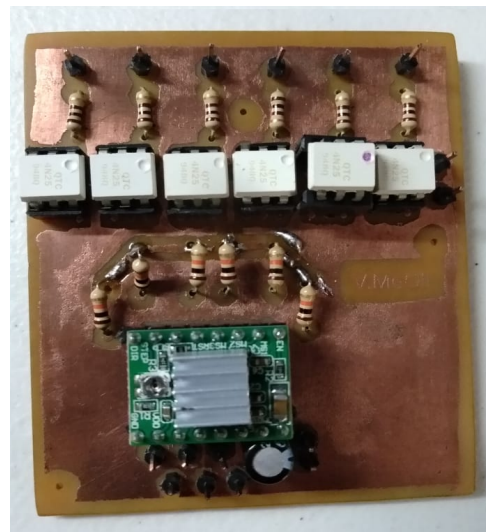


Figura 24 – PCB de acionamento com optocopladores

4.1.3 Interface

A interface GUI mostrou eficiência no que se propõe a fazer, todos os comandos foram bem enviados através da mesma e com resultados bem responsivos sobre a placa e sobre os componentes como a PiCamera.

A Figura 25 demonstra o resultado final da interface do usuário.

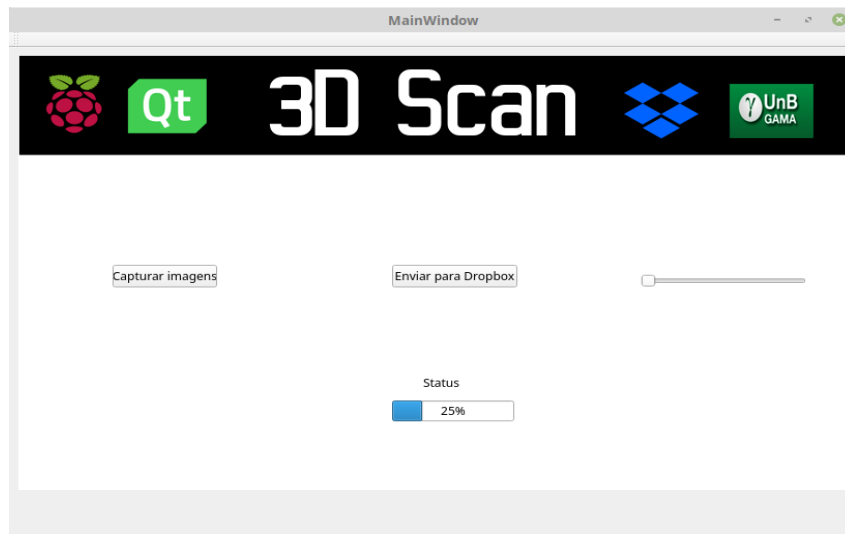


Figura 25 – Interface de usuário desenvolvida

4.1.4 Envio para Dropbox

O envio para o Dropbox foi feito com sucesso e as imagens podem ser usadas para aplicar a técnica de fotogrametria.

Softwares como o VisualSFM ou MeshRoom podem ser aplicados para que o modelo seja criado. Para melhor resolução e re-renderização pode se fazer útil o uso do Blender, principalmente na criação de modelos que possam ser imprimidos em impressoras 3D. Todos os softwares são gratuitos, mas em caso de estudante, existem licenças para o Recap da Autodesk.

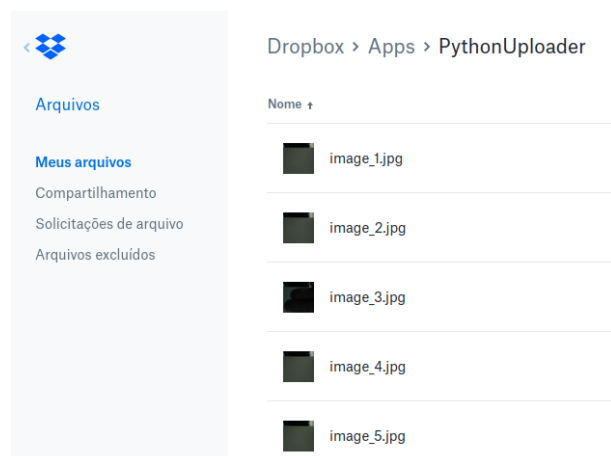


Figura 26 – Imagens enviadas para Dropbox

Todo o código para execução do projeto está em https://github.com/VitorMeirelesOliveira/scan_tcc.

4.1.5 Eficiência das imagens demonstrada em nuvem de pontos

Na figura 27, estão apresentados objetos a serem escaneados. O objeto da esquerda possui uma textura mais rígida e áspera e isso facilita a construção do modelo 3D. Há também a diferença na plataforma giratória, a da esquerda está com um adesivo com um padrão quadriculado para testar o perfil de captação e interpretação das imagens pelo software.

Para execução do *software* para gerar o modelo 3D, é necessário computador com placa gráfica dedicada devido a alta demanda de processamento gráfico, sendo impossível a execução de *softwares* com tamanha exigência, como os que foram usados, por computadores com placa de vídeo integrada. Caso seja usado o *software MeshRoom* ainda se faz mais exigência para marca nVidia, na qual o *software* foi projetado.

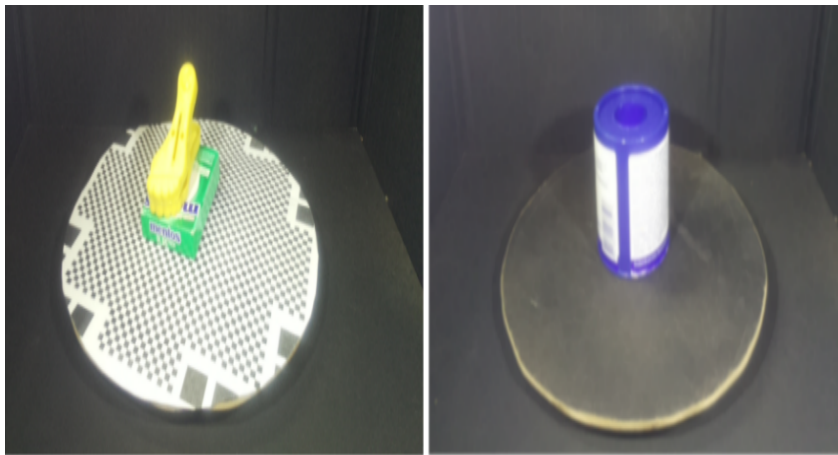


Figura 27 – Momento de captura de imagens para gerar o modelo 3D

O tempo de obtenção das imagens na memória da Raspberry Pi tem relação direta de ajuste com a Equação 3.1, pois quanto menor o ângulo para que cada foto seja capturada, mais fotos deverão ser obtidas para que todo o objeto seja contemplado por completo. A figura 28 demonstra os tetraedros representando a posição que a câmera está localizada espacialmente no momento da foto.

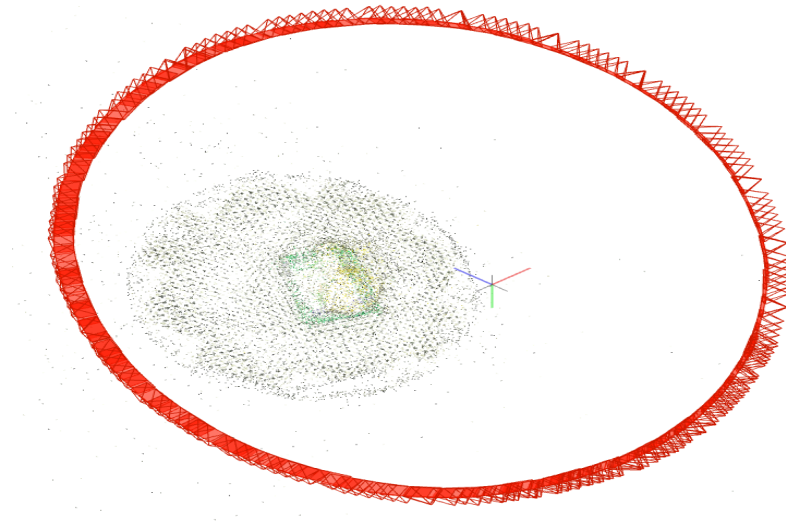


Figura 28 – Representação da posição espacial no momento que as fotos foram obtidas

A seguir, na Figura 29 está representado o resultado a partir do processamento pelo Software ColMap e visualizado e filtrado pelo software MeshLab. O processamento ocorreu em um Laptop da marca Avell com processador i7 da sétima geração 7700HQ, 16GB e com placa gráfica dedicada e moderna (GPU GTX 960).

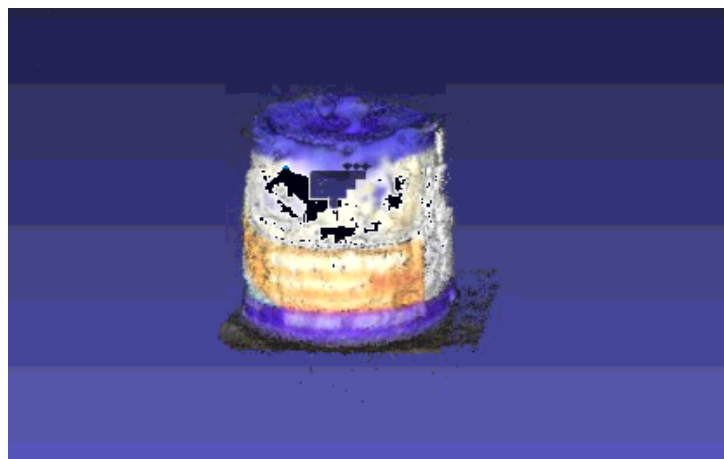


Figura 29 – Modelo reconstruído com filtros do software *MeshLab*

Para execução dos processos para criação do modelo 3D da Figura 30 foram exigidos 7 minutos para captura de 120 imagens a cada 3°. O envio das imagens para a nuvem leva cerca de 20 minutos. As imagens foram processadas em um computador da marca Avell, com 16Gb de memória Ram, processador core i7 HQ e placa gráfica dedicada nVidia 960GTX, no *software* VisualSFM e a limpeza da nuvem com o MeshLab resulta na imagem da direita da Figura 30.

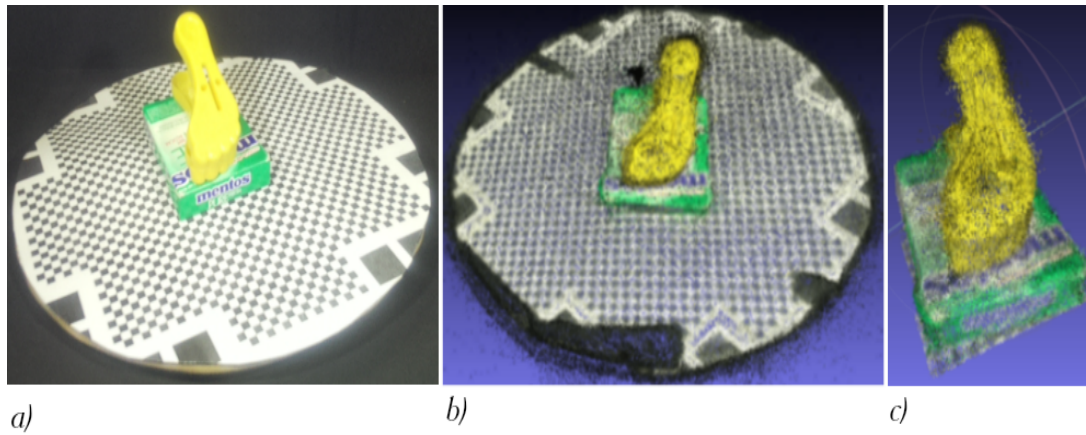


Figura 30 – Nuvem de pontos comprovando o resultado e eficiência da captura das imagens do modelo 3D. a) Modelo original. b) Modelo em nuvem de pontos gerado. c) Modelo em nuvem de pontos gerado e tratado

5 Conclusões

Este trabalho consiste no desenvolvimento de uma plataforma giratória para facilitar o trabalho de escaneamento 3D de objetos utilizando a técnica de fotogrametria. Para comprovar a eficiência do processo com as imagens coletados com a tecnologia embarcada, nuvens de pontos são geradas em computador pessoal utilizando *softwares* específicos.

Para a execução do trabalho é proposto o uso de um computador em placa, Raspberry Pi 3, acoplado com um *display touch* para que o usuário possa enviar as imagens capturadas de forma automática para o seu Dropbox e assim aplicar o uso de softwares específicos para que o modelo 3D seja gerado.

A implementação da interface gráfica se deu através de um *framework* que faz uso de C++ e as ações que executam por fundo são feitas em Python 3.

Como resultados obteve-se sucesso quanto à automatização dos movimentos e captura das imagens pela PiCamera V2, a câmera acoplada ao Raspberry Pi. Assim como o sucesso no envio das fotos por rede ao Dropbox e também no funcionamento seguro das ações evitando risco de queimar dispositivos através do uso de optoacopladores.

A criação do protótipo tomou forma com a usinagem da estrutura, quanto a parte eletrônica, a junção da interface com o tempo de execução das tarefas foi o ponto importante para execução dos processos do trabalho. A solução em iluminação para destaque do objeto se deu com o uso de LED's automotivos que fizeram destaque do objeto ao centro da plataforma giratória facilitando as fotografias da câmera e principalmente o tratamento posterior por software, o melhor desempenho foi sempre em luminosidade máxima.

A criação da nuvem de pontos através de *software* foi um ponto determinante para comprovar a eficiência do projeto. Sem que fosse possível gerar um modelo 3D o objetivo não teria sido alcançado.

Por fim, é inferido que para trabalhos futuros com a estrutura feita e com o envio sendo possível, seria viável melhoramento na qualidade das fotos já tiradas, passando por uma pre-filtragem, assim como a automatização dos cliques nos softwares utilizados. Também é interessante destacar situações de segurança, caso o usuário tome alguma atitude errada no uso do equipamento.

Referências

- 3DFLOW. *3DF Zephyr Photography guide*. 2018. Disponível em: <<https://www.3dflow.net/technology/documents/photogrammetry-how-to-acquire-pictures/>>. Acesso em: 6.12.2018. Citado 2 vezes nas páginas 15 e 28.
- AGARWAL LARS ARGE, A. D. P. K. From point cloud to grid dem: A scalable approach. 2018. Citado na página 25.
- ALICEVISION. *AliceVision - Photogrammetric Computer Vision Framework*. 2018. Disponível em: <<https://github.com/alicevision/AliceVision>>. Acesso em: 6.12.2018. Citado na página 31.
- ALLEGRO. Dmos microstepping driver with translator and overcurrent protection. 2014. Citado na página 36.
- BASRI, R.; JACOS, D. W. Lambertian reflectance and linear subspaces. 2003. Citado na página 23.
- BLENDER. *Blender*. 2018. Disponível em: <<https://www.blender.org>>. Acesso em: 6.12.2018. Citado na página 32.
- COLMAP. *Colmap*. 2018. Disponível em: <<https://colmap.github.io>>. Acesso em: 6.12.2018. Citado na página 31.
- DROPBOX. *Dropbox*. 2018. Disponível em: <<https://www.dropbox.com/about>>. Acesso em: 1.12.2018. Citado na página 40.
- EETIMES. *The why, where and what of low-power Soc design*. 2018. Disponível em: <https://www.eetimes.com/document.asp?doc_id=1276973>. Acesso em: 6.12.2018. Citado na página 30.
- EMBARCADOS. *Sistema Embarcado - O que é? Qual a sua importância?* 2018. Disponível em: <<https://www.embarcados.com.br/sistema-embarcado/>>. Acesso em: 6.12.2018. Citado na página 28.
- GARTNER. *Gartner Says Cloud Computing Will Be As Influential As E-business*. 2018. Disponível em: <<https://www.gartner.com/newsroom/id/707508>>. Acesso em: 6.12.2018. Citado na página 40.
- HOPE, C. *Command line vs GUI*. 2018. Disponível em: <<https://www.computerhope.com/issues/ch000619.htm>>. Acesso em: 1.12.2018. Citado na página 33.
- MACHINE, W. *STEREOSCOPIIC VISION AND PERSPECTIVE PROJECTION*. 2018. Disponível em: <<https://web.archive.org/web/20080920173058/http://www.cogs.susx.ac.uk/users/davidy/teachvision/vision5.html>>. Acesso em: 1.12.2018. Citado na página 27.
- MAKE. *Heavy Duty Turntable for 3D Scanning People*. 2015. Disponível em: <<https://makezine.com/projects/guide-to-3d-printing-2014/heavy-duty-turntable/>>. Acesso em: 28.11.2017. Citado 2 vezes nas páginas 25 e 26.

- MESHLAB. *MeshLab*. 2018. Disponível em: <<http://www.meshlab.net>>. Acesso em: 6.12.2018. Citado na página 32.
- MITTAL, S. A survey of fpga-based accelerators for convolutional neural networks. *Neural Computing and Applications*, 09 2018. Citado na página 30.
- OLIVEIRA, S. de. *Internet das coisas com ESP8266, Arduino e Raspberry Pi*. São Paulo: Novatec, 2017. Citado 2 vezes nas páginas 23 e 29.
- RASPBERRY. *Raspberry Pi 3 Model B+*. 2018. Disponível em: <<https://www.raspberrypi.org/products/raspberry-pi-3-model-b-plus/>>. Acesso em: 1.12.2018. Citado na página 34.
- REUSE, D. *Power Management for Internet of Things(IoT) System on a Chip(SoC) Development*. 2018. Disponível em: <<https://www.design-reuse.com/articles/42705/power-management-for-iot-soc-development.html>>. Acesso em: 6.12.2018. Citado na página 30.
- RUSHMEIER, F. B. e H. 3d model aquisition pipeline. 2002. Citado na página 26.
- SITNIK MACIEJ KARASZEWSKI, W. Z. R.; RUTKIEWICZ, J. Automation of 3d scan data capturing and processing. 2010. Citado 2 vezes nas páginas 23 e 25.
- SPARKFUN. *Pulse Width Modulation*. 2018. Disponível em: <<https://learn.sparkfun.com/tutorials/pulse-width-modulation/all>>. Acesso em: 1.12.2018. Citado 3 vezes nas páginas 15, 39 e 40.
- SUŽIEDELYTÈ-VISOCKIENÈ; JŪRATÈ; BAGDŽIŪNAITÈ. Close-range photogrammetry enables documentation of environment-induced deformation of architectural heritage. *Environmental engineering and management journal*, v. 14, p. 1371–1381, 07 2015. Citado na página 27.
- TORONTO, U. of. *FPGA Architecture for the Challenge*. 2018. Disponível em: <http://www.eecg.toronto.edu/~vaughn/challenge/fpga_arch.html>. Acesso em: 6.12.2018. Citado na página 30.
- WU, C. *VisualSFM : A Visual Structure from Motion System*. 2018. Disponível em: <<http://ccwu.me/vsfm/>>. Acesso em: 6.12.2018. Citado na página 31.