

# SENSOR DE DISTÂNCIA UTILIZANDO FPGA, CÂMERA E LASER

**Guilherme Lopes Wanderley<sup>1</sup>; Anfranserai Morais Dias<sup>2</sup>**

1. Bolsista PROBIC/UEFS, Graduando em Engenharia da Computação, Universidade Estadual de Feira de Santana, e-mail: guilhermelw.uefs@gmail.com

2. Orientador, Departamento de Tecnologia, Universidade Estadual de Feira de Santana, e-mail: anfranserai@ecomp.uefs.br

**PALAVRAS-CHAVE:** sensor de distância; visão computacional; FPGA.

## INTRODUÇÃO

Este resumo trata do desenvolvimento de um sensor de distância utilizando um *Field Programmable Gate Array* (FPGA), uma câmera e um laser de ponto. A construção do sensor faz parte do projeto de pesquisa Construção de uma colônia de robôs autônomos para reconhecimento, busca e inspeção. Neste sentido, medir as distâncias é uma atividade fundamental tanto para navegação - saber a distância dos obstáculos - quanto para mapeamento do local - desenhar um mapa com distâncias acuradas. Em seguida são discutidas as metodologias na construção e calibragem do sensor.

## MATERIAL E MÉTODOS OU METODOLOGIA (ou equivalente)

O processo de construção do sensor foi dividido em três partes: modelagem, implementação e calibragem. A modelagem foi realizada pelo trabalho anterior (Jesus, 2015), portanto apenas serão citados os pontos em que esta foi modificada devido à dificuldades na implementação ou calibragem do sensor.

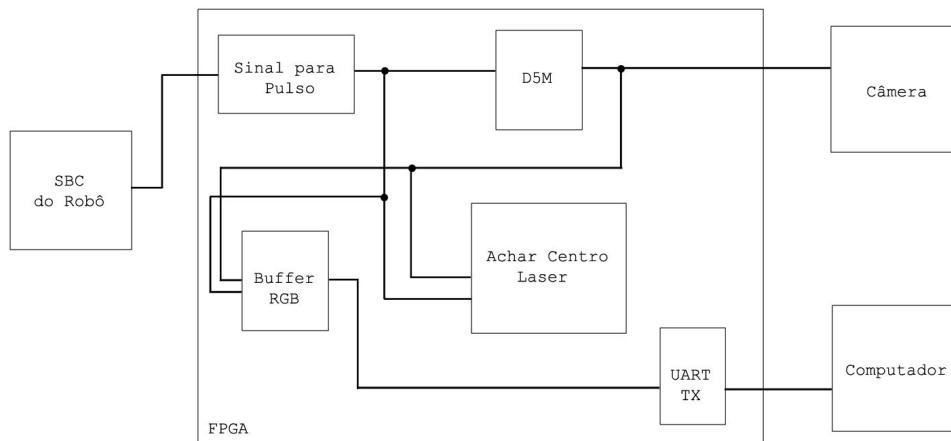
### Modelagem

A modelagem do sistema sofreu modificações devido a problemas no armazenamento da imagem na *synchronous dynamic random-access memory* (SDRAM). Por não conseguir utilizar a memória, a aquisição do valor do *threshold* do laser foi modificada. Outrora obtido pela subtração de duas imagens em escala de cinza (uma contendo o laser e outra não), o valor do *threshold* passou a ser obtido utilizando a técnica de mineração de dados chamada Agrupamento Visual a partir de diversas amostras de imagem com laser. Esta técnica consiste em utilizar o processamento do computador a fim de reduzir a dimensionalidade dos dados e permitir ao usuário humano distinguir claramente os grupos semânticos distintos presentes nos dados (Aggarwal, 2015). Trinta amostras foram obtidas através da captura de imagens por meio da câmera de trabalho em diferentes horários e condições de iluminação, para que o sensor consiga distinguir o laser da imagem em diferentes condições, aumentando a robustez do sistema. Foram coletados, para cada pixel, os dados de: posição no eixo X e Y da imagem e valores de cada cor (vermelho, verde e azul) em profundidade de 12 bits, determinada pela câmera.

### Implementação

A implementação do sensor envolve os seguintes componentes: a câmera OV7670, presente no kit visual TRDM-D5M, o kit de desenvolvimento DE0-Nano (distribuído

pela Intel FPGA) contendo o *Field Programmable Gate Array* (FPGA) Cyclone IV EP4CE22F17C6N, além do componente eletrônico laser de ponto. A Figura 1 mostra o diagrama de blocos do sistema. O computador do laboratório é utilizado para validação dos algoritmos, enquanto o *Single Board Computer* (SBC) do robô é o destino final da medição da distância. O *hardware* de interesse para este trabalho é o FPGA, pois nele são implementados os circuitos digitais responsáveis pelo controle da câmera e pelo cálculo da distância. Em seguida, são discutidos os blocos de circuito do FPGA.



**Figura 1:** Diagrama de blocos do sistema

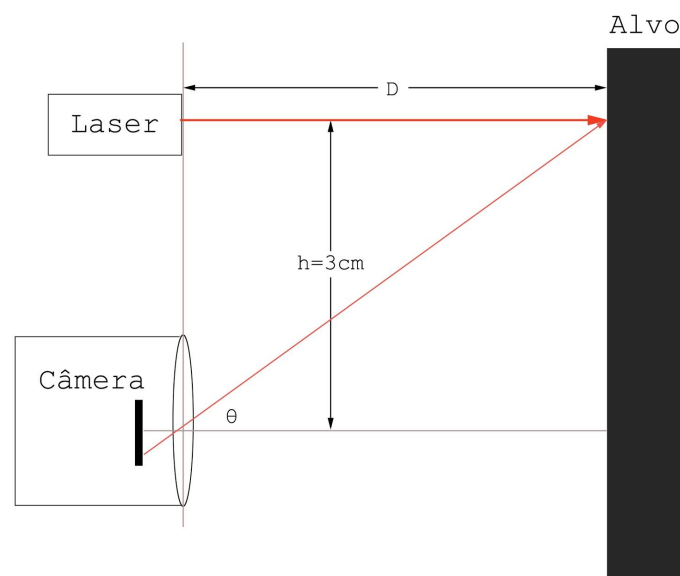
O primeiro bloco, SinalParaPulso, é responsável por gerar um pulso de captura, chamado *trigger* (gatilho), para a câmera. O bloco D5M é o responsável pela comunicação com a câmera, tanto envio quanto recebimento de dados. Este bloco é disponibilizado pela Intel FPGA. É este bloco que contém as configurações iniciais a serem lidas pela câmera. Também é este bloco que recebe a imagem da câmera.

Para que o computador consiga montar a imagem na ordem correta, é necessário primeiro armazenar estes pixels no FPGA. O bloco BufferRGB cumpre este papel, recebendo os pixels do bloco D5M a partir de uma determinada linha e armazenando-os na memória do FPGA. Assim, com o trecho de imagem armazenado no FPGA, é possível enviá-lo ao computador na velocidade correta, para que a imagem seja corretamente montada. Para aumentar o tamanho físico do trecho armazenado, são armazenados 16 bits para cada pixel de acordo com o padrão de cores RGB-565. Cinco bits para as cores vermelho e azul, seis bits para a cor verde. Quando este buffer estiver cheio, ele sinaliza ao bloco UARTtx para começar a transmitir seus dados para o computador. Este bloco transmite a imagem ao computador de oito em oito bits, de acordo com o padrão RS-232.

O último bloco desenvolvido foi o AcharCentroLaser, responsável por calcular as coordenadas centrais do laser na imagem. Este circuito possui como entrada os pixels lidos pela câmera e determina se o pixel recebido contém ou não o laser, de acordo com o valor de threshold estabelecido. Caso contenha, o valor de altura do laser é incrementado ao achar o laser em uma nova linha. Após processados todos os pixels da imagem, subtrai-se da altura da última linha em que o laser foi capturado a metade da altura do laser, obtendo-se assim a altura do centro do laser na imagem.

### Calibragem

Depois de calculada a coordenada do ponto central do *laser* na imagem, foi iniciado o processo de calibragem. Dado as configurações mostradas pela Figura 2, é necessário obter os valores para  $rpc$  e  $ro$ , tal que  $\theta = h/D$  e  $\theta = pfc*rpc + ro$  como indicado por (Jesus, 2015). Estes valores são obtidos através de uma regressão, neste caso linear. Esta não é a regressão mais acurada, porém tanto a modelagem quanto os experimentos utilizando esta regressão mostraram performance satisfatória, o que simplifica o modelo a ser embarcado. Esta regressão é obtida utilizando diversas imagens a diferentes distâncias conhecidas, obtendo um valor de Teta ( $\theta$ ) para cada uma. Em seguida, os valores de  $\theta$  e  $pfc$  são utilizados na regressão para determinar os valores de  $rpc$  e  $ro$ .



**Figura 2:** Representação do cenário da distância a ser calculada

O cálculo embarcado da distância não foi obtido, porém está a poucas horas de trabalho de sua implementação. Ainda assim, o cálculo da distância foi realizado em *software* a partir de medidas do sensor, o que possibilita discussões fiéis sobre precisão e acurácia dos resultados.

### RESULTADOS E/OU DISCUSSÃO (ou Análise e discussão dos resultados)

Quase todos os circuitos propostos na metodologia foram construídos e sua correteude foi assegurada, excetuando-se o circuito final de cálculo da distância, pois preferiu-se calibrar e estudar o sensor a embarcá-lo de forma imprecisa.

O valor de *threshold* do laser foi definido como 3072 considerando um inteiro de 12 bits - profundidade das cores da câmera. Isto é, caso os dois bits mais significativos do vermelho obtido pela câmera possuam valor lógico 1, o pixel em questão possui uma imagem de laser. Após definido o *threshold*, através do sensor foram medidas as alturas do laser na imagem para cada posição de distância. Com estes valores, foi realizada a regressão linear, obtendo os valores de 0,0007 pra *rpc* e -0,3978 *ro*. O cálculo final da distância foi realizado em *software*, obtendo-se os resultados exibidos na Tabela 1.

Tabela 1. Relação entre distância a ser medida e altura do laser na imagem

<b>Distância real (cm)</b>	<b>Altura do laser na imagem (pixels)</b>	<b>Teta <math>\theta</math> (radianos)</b>	<b>Distância calculada (cm)</b>	<b>Erro absoluto (cm)</b>	<b>Erro relativo (%)</b>
20	743	0,1488899476	20,163	-0,163	0,815
30	680	0,0996686525	29,454	0,546	1,820
40	643	0,0748598477	40,264	-0,264	0,660
50	624	0,0599281551	49,578	0,422	0,844
60	611	0,0499583957	58,885	1,115	1,858
70	600	0,0428309327	69,991	0,009	0,013
80	592	0,0374824367	81,111	-1,111	1,389
90	586	0,0333209959	92,079	-2,079	2,310
100	582	0,0299910049	101,200	-1,2	1,200

Estes resultados demonstram a acurácia do laser. Apenas a medida para a distância de 90 cm teve erro absoluto maior que 1,2 centímetros e erro relativo maior que 2%. Estas medidas são satisfatórias até mesmo para a atividade de mapeamento, uma vez que o robô, em geral, trabalha com blocos de 20x20cm. Um erro desta magnitude não provocaria uma falha no sistema de navegação e mapeamento.

### **CONSIDERAÇÕES FINAIS (ou Conclusão)**

Construir um sensor optoeletrônico não é trivial. Após toda a modelagem e construção correta dos circuitos, vários ajustes finos são necessários e diversos fatores podem influenciar na medida do sensor. O trabalho atual implementa estes circuitos e identifica os fatores que os influenciam. Além disto, são propostas otimizações para o circuito a fim de aumentar a acurácia da medida e aumentar a robustez do sensor às variações das condições, principalmente quanto à distância e à iluminação. O sensor foi construído, testado e sua precisão e acurácia foram verificadas. Os próximos passos são embarcar o cálculo final e realizar a comunicação com o robô.

### **REFERÊNCIAS**

JESUS, F.B; JESUS, T.C. 2015. Projeto de sensor de distância baseado em laser utilizando FPGA. *In: XIX Seminário de Iniciação Científica da UEFS, Feira de Santana.*  
 AGGARWAL, C.C; 2015. *Data Mining: The textbook.* Springer.

