

Ponto de Controle 3

Leitor de Códigos de Barras para Carrinhos de Compras

Erick Antonio Correa dos Reis - 150034156
Universidade de Brasília
St. Leste Projeção A - Gama - DF, 72444-240
email: erickcorreareis@gmail.com

Tiago Avelino Ribeiro da Silva - 150022662
Universidade de Brasília
St. Leste Projeção A - Gama - DF, 72444-240
email: tiago.avelino1997@gmail.com

Resumo - Leitor de códigos de barras para carrinhos de compras utilizando câmera.

Palavras-chave - Códigos de barras; Raspberry Pi; Processamento de imagens.

I. INTRODUÇÃO

Algumas atividades comuns do dia-a-dia costumam demandar muito tempo, um exemplo clássico são as filas de supermercados, mesmo depois de um longo tempo escolhendo seus produtos ainda é necessário enfrentar demoradas filas para que sejam conferidos todos seus produtos.

Visando reduzir o tempo gasto nessa atividade, foi pensado em um dispositivo integrado ao carro de compras que permite analisar o preço do produto a ser comprado, ao mesmo tempo que permite acrescentar o preço do produto, aos demais do carro. Será feita a leitura do código de barras por meio de uma câmera, lendo os símbolos deste código, pela variação na largura das barras e assim interpretando o produto a ser comprado, e o valor deste.

II. OBJETIVOS

Implementar um sistema que mostre o valor total e atualizado dos itens que estão no carrinho através de um leitor de códigos de barras por câmera.

III. JUSTIFICATIVA

Estabelecimentos que não utilizam dessa técnica já são bastante comuns na Europa por exemplo, onde os próprios clientes passam seus produtos no caixa e realizam o pagamento, ainda assim é necessário certo trabalho para registrar todos os itens que já estavam no carrinho de compras.

Buscando resolver este problema, o projeto em questão visa implementar um sistema que mostra o valor total da compra em um display no próprio carrinho, esse valor também

deve ser atualizado assim que um item é adicionado ou retirado do carrinho, fazendo com o pagamento ao final da compra seja muito mais rápido.

A utilização de uma câmera para a identificação do código do produto é algo que deve facilitar bastante a utilização do sistema já que o processamento de imagens em uma placa como a Raspberry Pi 3 é bastante rápido.

IV. REQUISITOS

Como requisitos do sistemas, visa-se obter:

- Decodificar com eficiência os códigos de barras.
- Registrar imagens com boa nitidez para facilitar a decodificação.
- Registrar os itens e atualizar os valor no display em um tempo limite: 2 segundos.
- Fazer a soma dos valores das compras em tempo real.

V. DESCRIÇÃO DE HARDWARE

O projeto tem como foco o processamento de imagens, dessa forma não há uma ênfase em hardware, tendo em vista que boa parte das análises são realizadas para o entendimento e o desenvolvimento em software. Dessa forma o hardware do sistema se concentra em um display de LCD nokia 5110, que visa fazer a interface entre usuário e hardware, aliada há um teclado matricial personalizado, que realiza todas as funcionalidades do sistema.

A. Interface para o usuário.

Em hardware será feita a interface para o usuário onde este poderá colocar o produto a ser comprado no carrinho, ou caso tenha colocado um produto erroneamente retirar tal produto e ainda visualizar sua lista de compras já realizada. Essa atividade só será possível por meio do teclado matricial, que apresentará botões de avançar nas telas onde está sendo realizada a compra e caso necessário retirar o produto selecionado no display.

Foi realizada a codificação da comunicação spi com o display do nokia 5110 através da biblioteca armbianIO. Após a realização da comunicação foram criadas as threads que visam promover as diversas telas que serão utilizadas, sendo essas as telas de inicialização mostrando a data e hora do momento da compra, ainda nessa tela foi realizada a chamada do “botão começar” para entrar na próxima tela.

Na tela seguinte é colocado o último produto a ser comprado sendo que nessa tela há uma mensagem que requisita o botão “adicionar”, para direcionar a tela inicialização da câmera, assim a câmera só é ativada no momento de leitura do código de barras, visando poupar energia do sistema. Após a leitura do produto há o pedido da quantidade de produtos do mesmo tipo sendo comprados, para economizar tempo.

Quando é acionado o botão para realizar a compra é feita uma nova mudança de tela, e a câmera é ativada, lendo o código de barras do produto, caso em um período de tempo pré definido não seja possível realizar a leitura, o sistema alerta ao usuário com uma mensagem nesta tela que o botão seja pressionado novamente para realizar a leitura ou para voltar para tela onde o último item comprado está presente.

Caso um dos itens seja adicionado erroneamente há um botão que redireciona o usuário a lista completa de itens comprados, e nessa nova tela há um botão de selecionar o produto, subindo ou descendo na lista, com o produto selecionado bastará apertar o botão de retirar o produto.

Em cada tela há uma thread de gerenciamento, que determina quais as operações que o usuário pode realizar, sendo que em alguns casos é necessário habilitar a interrupção do botão. Na thread principal apenas são somados os valores de cada produto, associada a sua quantidade

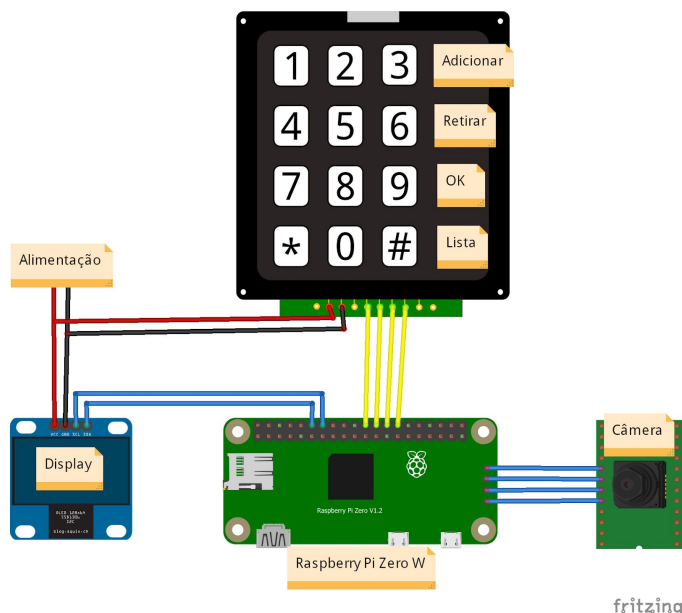


Figura 1. Esquemático do hardware

VI. DESCRIÇÃO DE SOFTWARE

Para implementar o projeto é utilizada a biblioteca OpenCv[3] - *Open Source Computer Vision Library*, essa biblioteca tem como propósito auxiliar na construção de projetos na área de visão computacional, possuindo módulos para o tratamento de fotos e vídeos, utilizando a linguagem Python. Foi implementado inicialmente o projeto em linux para verificar a funcionalidade desta em seguida os blocos de códigos da biblioteca a serem utilizados serão implementados na Raspberry Pi. Serão explicados os blocos de códigos utilizados.

A. Identificador de Qrcode e barcode.

Para resolver esse requisito foi utilizada a abordagem de cascade, que consiste em determinar retângulos em volta da figura analisada (código de barras ou Qrcode) e após a identificação da área onde será feito o processamento de imagem, fazer a decodificação desse código e mostrar esse código decifrado no terminal, foi dado o nome de Qrcode.py para este código.

B. Cadastro dos produtos.

Através do código anterior, será feito a união do código de barra relacionando este com o valor de mercado do produto, caso não haja esse produto será separada uma área de cadastro, prévio, realizado inicialmente pelos integrantes do projeto ou pelo usuário que tiver acesso a essa área (gerente ou funcionário da loja).

C. Valor efetivo.

Será realizada a soma de todos os valores adicionados ao carrinho e esta soma será mostrada no display Oled, assim como será gerada toda a lista dos produtos correspondentes à compra realizada.

VII. RESULTADOS

Para o ponto de controle atual foram obtidos os seguintes resultados:

- Um código que faz a comunicação com o display.
- Um código de controle de telas.
- Um código para tirar fotos.

Para o próximo ponto de controle os objetivos são unir os códigos em hardware com o código de decifragem do código de barras, visando a otimização e ainda realizar a otimização utilizando threads ou múltiplos processos na Raspberry Pi.

Além disso será realizada a construção do equipamento real que será equipado ao carrinho de compras do supermercado, dessa forma facilitando a utilização do projeto.

VIII. CONCLUSÃO

Tendo em visto o conhecimento adquirido até o atual momento do curso e as aulas a serem ministradas, é possível perceber que a proposta apresentada está adequada para a disciplina, sendo possível a implementação no tempo disponível para a realização do projeto.

Para o atual ponto de controle ainda não foi possível realizar a captura do código de barras tendo em vista que inicialmente seria utilizada uma webcam comum, porém foi realizada a compra da camera especifica para a raspberry pi, essa câmera apresenta uma comunicação um pouco diferente, sendo que a biblioteca opencv atual não dá o suporte exato necessário, tendo que mesclar a biblioteca Zbar dessa biblioteca com a biblioteca Pi cam, sendo que esta última ainda não havíamos nos familiarizado pela modificação da câmera que iríamos utilizar.

Dessa forma após realizar a comunicação com a câmera será necessário utilizar essas bibliotecas para a leitura do código e armazenar esse código em um arquivo que será lido para fazer a comparação com o banco de dados de produtos do supermercado.

REFERÊNCIAS

[1] Contagem de objetos em movimento com OpenCV e Python usando Raspberry Pi. Disponível em: <<https://www.embarcados.com.br/objetos-opencv-e-python-raspberry-pi/>>. Acesso em 04 abr. 2018.

[2] SIMÕES, Eduardo Dusanoski. DESENVOLVIMENTO DE SISTEMA PARA LEITURA DE CÓDIGO DE BARRAS COM "FEEDBACK" PARA AQUISIÇÃO E SEGURANÇA DE PRODUTOS EM SUPERMERCADOS. 2015. 53 f. Disponível em: <http://repositorio.roca.utfpr.edu.br/jspui/bitstream/1/6789/1/CT_CO_EAU_2015_1_10.pdf>. Acesso em: 04 abr. 2018

[3] OpenCv library. Acesso em 27 abr.2018 <<http://projectabstracts.com/list-of-projects-on-image-processing>>.

[4] ArbianIO library. Acesso em 26 mai.2018 às 9:00 horas <<https://github.com/bitbank2/ArmbianIO>>.