

PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS
Instituto de Ciências Exatas e de Informática

Mateus Loures do Nascimento
Pedro Augusto Prosdocimi Resende
Pedro Henrique Mattiello

Trabalho Prático:
A Implementação de um leitor de código de barra

Belo Horizonte
2019

Trabalho Prático:
A Implementação de um leitor de código de barra

Artigo apresentado para a disciplina de
Processamento de Imagens pelo Curso de
Ciência da Computação da Pontifícia
Universidade Católica de Minas Gerais –
PUC-MG

Professor: Alexei Manso Correa Machado

Belo Horizonte
2019

1 INTRODUÇÃO

Código de barras é uma representação visual de informações na forma de barras e espaços em uma superfície. As barras e espaços podem ter larguras diferentes e consistem em números, caracteres e símbolos. Diferentes combinações desses números e caracteres alfanuméricos são usadas para representar informações. Existem vários tipos de códigos de barras em uso atualmente, como o EAN-13, UPC-A, Código 128, Código 39 e entre outros.

Hoje os códigos de barras são amplamente utilizados em todas as mercadorias de varejo no intuito de manter o controle dos produtos disponíveis e facilitar o checkout dos mesmos. Atualmente, a maioria dos scanners de código de barras está usando métodos de infravermelho para escanear um código de barras. Isso pode levar ao problema de custo, onde esses scanners são caros e inacessíveis para o usuário. Para superar este problema, o sistema baseado em câmera para leitura de código de barras é aplicado para desenvolver um sistema de leitura de código de barras.

2 TÉCNICAS IMPLEMENTADAS PARA A SOLUÇÃO

O sistema usa algumas técnicas básicas de processamento de imagens, como a transformada de Hough e detecção de borda sub-pixel para o processamento da imagem de código de barras capturada por câmera.

Para fazer a segmentação a imagem foi transformada em tons de cinza e binarizada. Depois foi feita uma operação chamada BLACKHAT, que corresponde a pegar o resultado da operação de fechamento e subtrair da imagem original, isto desenha apenas as bordas pretas da imagem, como demonstrado abaixo:



Logo depois, foi utilizada a transformada de Hough para detectar as linhas da imagem. O local onde tem o maior número de linhas paralelas é entendido como o código de barras. Com uma dessas linhas é calculado a inclinação, e por essa inclinação é calculado a inclinação do código de barras.

Com este ângulo a imagem é rotacionada, de modo que o código fique na posição horizontal. Depois, é realizada operação de dilatação e fechamento, apenas na horizontal, e ao fim é aplicado uma operação de abertura também com ênfase na horizontal para reunir as linhas horizontais próximas, como mostrado na figura abaixo.

Por fim, é feita uma operação de abertura com uma máscara maior, também com ênfase horizontal, para reunir as áreas com muitas linhas.



Finalmente é desenhado um destaque em azul do código de barras.

Um código de barras 1D é lido horizontalmente, da esquerda para a direita. As informações armazenadas em um código de barras 1D são alfanuméricas (letras e números) e podem incluir informações como números de produtos ou um endereço. EAN-13 é a simbologia de código de barras linear mais popular que é usada em quase

todos os produtos de varejo. Na figura abaixo o código de barras é apresentado e explicado. Na parte da leitura do código de barra foi pensando uma lógica que retirava uma linha horizontal de pixels, média as barras iniciais e admitindo essa espessura como um bit e a partir daí se dava a leitura dos números. Mas não foi possível ler os números dessa forma por isso não foi mostrado os resultados.

Este sistema pode ser usado a qualquer momento e em qualquer lugar pelo usuário que gosta de digitalizar os números de código de barras sem ir a qualquer lugar que ofereça os serviços de scanner de código de barras.



3 RESULTADOS E LIMITAÇÕES

Em decorrência da grande maioria das imagens apresentarem leves distorções (sombra, rotação) os valores da decodificação do código para valor numérico apresentaram falhas de tradução, optamos por deixar a decodificação de lado e focar em desenvolver nosso código de detecção. (para o maior número de situações possíveis)

Como o algoritmo não faz a leitura e apenas a segmentação do código de barras de um modo bem simples o tempo foi menor que 1s. A segmentação pode apresentar problema se o código de barras estiver muito grande, já que na hora de reunir os pixels eles estarão distantes ou se existir muitas linhas paralelas, o programa irá pegar o ângulo incorreto. A rotação pode variar para algo em torno de 20% sem afetar na detecção das barras.

4 BIBLIOTECAS UTILIZADAS

A biblioteca utilizada nesse sistema foi o OpenCV. *Open Source Computer Vision Library* é uma biblioteca de software de visão computacional e de *machine learning* em código aberto. Essa biblioteca foi construída para fornecer uma infraestrutura comum para aplicativos de visão computacional e para acelerar o uso da percepção da máquina nos produtos comerciais.

O OpenCV tem mais de 2500 algoritmos otimizados, o que inclui um conjunto abrangente de algoritmos de visão computacional e de aprendizado de máquina clássicos e de última geração. Esses algoritmos podem ser usados para detectar e reconhecer rostos, identificar objetos, classificar ações humanas em vídeos, rastrear movimentos de câmera, rastrear objetos em movimento, extrair modelos 3D de objetos e entre outros.

O primeiro passo para instalar o OpenCV no projeto é realizar o download do site <https://opencv.org/releases/>. Após instalar a última versão da biblioteca é necessário extrair os arquivos em uma localização qualquer. Feito isso, basta importar o arquivo *opencv-3xx.jar* e a biblioteca *opencv_java3xx.dll* para o projeto para começar a utilizar seus recursos.