

# Algoritmo para Localização de uma Imagem Código

ALEX HOLZTRATNER<sup>1</sup>  
WALTER GODOY JR<sup>2</sup>  
OGÊ MARQUES FILHO<sup>2</sup>

<sup>1</sup>EQUITEL Equipamentos e Sistemas de Telecomunicações S.A.  
Rua AT-6 (Pedro Gusso), 2635  
81310-900 Curitiba, PR, Brasil

<sup>2</sup>CEFET PR - Centro Federal de Educação Tecnológica do Paraná  
Av. 7 de setembro, 3165  
80230-901 Curitiba, PR, Brasil

**Abstract.** This paper describes an algorithm to find a given image based in finding some characteristic sub areas. These sub areas are searched with a binary template matching technique that is optimized to be computationally fast. The algorithm is implemented and has an application at a wider project.

## 1. Introdução.\*

A localização de uma determinada imagem dentro de uma imagem maior é um tema que desafia muitos pesquisadores em todo mundo. Tarefa rotineira realizada pelo cérebro humano, para as máquinas ela é tanto mais complicada quanto maiores forem as variações nas dimensões, orientação e localização da imagem procurada.

Neste trabalho, procura-se limitar a dificuldade buscando imagens de dimensões e orientação fixas, limitando-se à localização da imagem. Todavia, como pretende-se uma aplicação prática para o trabalho, buscou-se tolerar pequenas distorções nas dimensões e leves rotações da imagem. Essas advêm do processo mecânico / ótico de leitura da imagem e não podem ser eliminadas.

A próxima seção descreve as características da imagem procurada e da imagem onde ela está contida. Os principais problemas para a localização da imagem, quase todos relacionados com o custo computacional do processo, são levantados. Segue-se com uma descrição do algoritmo implementado, salientando-se também as soluções adotadas para acelerar o processamento. São apresentadas as aplicações para o processo e por fim, tecidas conclusões.

## 2. Descrição do Problema.

Propõe-se encontrar uma imagem-código dentro da imagem de uma página que pode conter qualquer tipo de informação bi-tonal com predominância de branco. Esta é o tipo de imagem tipicamente transmitida por

sistemas fac-símile e contém texto escrito manual ou mecanicamente, desenhos, figuras ou uma combinação destes. Uma imagem fac-símile é normalmente obtida por leitura ótica linha a linha. Cada linha contém, nos modelos comuns, 1728 elementos de imagem (pels). Não existe definição de número mínimo ou máximo de linhas mas a resolução normal de leitura é de 3.85 linhas/mm o que nos fornece por volta de 1080 linhas em uma página A4.

A imagem-código, que é a que se pretende encontrar, constitui-se de um padrão de codificação qualquer para identificação de objetos que se deseje controlar como, por exemplo, o código de barras ou ainda a matriz de pontos.

Da natureza da imagem código surge o primeiro problema para a sua localização. Como se trata de um padrão para identificação, ele varia de acordo com o objeto identificado, não sendo sempre o mesmo. Para facilitar a localização, a imagem deve possuir características marcantes e sempre presentes que possam orientar a procura.

Uma análise superficial do problema pode levar a concluir que bastaria comparar uma cópia da imagem a ser encontrada com todas as posições possíveis na imagem que a contém. Mas ao tomarmos conhecimento que uma imagem de 1 por 1 cm já contém mais de 300 pontos a serem testados por volta de 1 milhão e 800 mil vezes para uma imagem de uma página, vemos que isso é inviável. Isso sem mencionar que nesse método, a menor distorção já impediria a localização da imagem.

As técnicas de pré-processamento de imagens para extrair seus elementos básicos também foram consideradas. Estas porém são computacionalmente caras e após a sua execução ainda deve-se reprocessar toda a imagem para encontrar a imagem-código desejada, voltando-se então à imagem original para extrair a

\* Este trabalho foi desenvolvido nas dependências do Centro Federal de Educação Tecnológica do Paraná com o apoio da EQUITEL Equipamentos e Sistemas de Telecomunicações S.A. e do Instituto Euvaldo Lodi do Paraná.

imagem encontrada. Todo esse processamento é lento e ainda é necessário maior espaço de armazenamento para guardar as fases intermediárias do processo.

### 3. Algoritmo.

Inicialmente, para possibilitar que um padrão de codificação pudesse ser localizado como uma imagem constante dentro da imagem maior, foi necessário acrescentar alguma característica fixa na imagem. Este acréscimo também não pode interferir na própria codificação que a imagem contém. Assim, foi decidido colocar uma moldura em torno da imagem. Essa moldura permite que o usuário oriente melhor a imagem código dentro da imagem maior. Localizada a moldura, automaticamente está localizada a imagem código.

Por motivos não relevantes a esse trabalho em particular, escolhemos como padrão de codificação a matriz de 7x7 pontos [Gils (1986)]. A figura 1 mostra um exemplo da imagem código que o algoritmo ora descrito localiza em uma imagem maior.

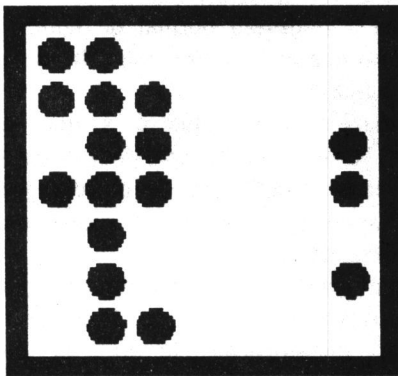


Figura 1: Imagem código.

Da imagem código tomou-se quatro pontos característicos como indicadores da presença da borda. Escolheram-se os quatro cantos internos como sendo os mais significativos. Note-se que o algoritmo de localização pode ser aplicado a outras imagens, bastando-se para isso identificar regiões bem marcantes da imagem, para, detectada a sua presença, assumirmos que a imagem foi encontrada. Estas regiões devem ser pequenas dentro da resolução da imagem maior. O algoritmo localiza regiões com dimensões de 8x8 pels. Devemos escolher um mínimo de três regiões para que haja confiabilidade na localização, sendo o algoritmo tanto mais seguro quanto maior for o número de regiões. Obviamente, um número muito grande de regiões torna a localização mais demorada.

Para localizar as regiões dentro da imagem utiliza-se um processo chamado de correlação com máscaras adaptado para imagens bitonais [Marques (1990)]. A seguir descreve-se essa técnica.

#### 3.1 Correlação com máscaras.

Neste processo, cada região de 8x8 pels da imagem processada é comparada com uma máscara que descreve a região que se quer localizar. Essa máscara descreve a região a ser localizada através da definição de quais pels devem ser claros, quais escuros e quais podem ser tanto claros quanto escuros. Os pels que podem ser tanto claros quanto escuros implementam a tolerância da máscara à distorções na imagem e são chamados de pels indeterminados. Para então preparar a máscara utilizada para a correlação, deve-se selecionar uma fração da imagem, de dimensões 8x8 pels que contenha aproximadamente o mesmo número de pels claros e escuros. Essa fração deve obviamente conter a característica procurada. Os pels que margeiam a linha que separa a região clara da escura são assim os mais sujeitos a trocar de tom devido a quantização realizada no momento da leitura ótica da imagem. Define-se assim na máscara esses pels como indeterminados. Os pels restantes são fixados na máscara como claros ou escuros de acordo com o que existe na fração selecionada para a construção da máscara. A figura 2 mostra um exemplo de máscara utilizada para localizar o canto superior esquerdo da moldura da imagem. O canto da moldura, mais especificamente o vértice interno dele foi escolhido pois é uma característica marcante da imagem código.

```

0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 X X X X X X
0 0 X 1 1 1 1 1
0 0 X 1 1 1 1 1
0 0 X 1 1 1 1 1
0 0 X 1 1 1 1 1
0 0 X 1 1 1 1 1

```

Figura 2: Máscara para localizar canto superior esquerdo da moldura (0 = escuro, 1 = claro e X = tanto claro quanto escuro).

A máscara definida é armazenada em dois conjuntos de 8 bytes. Cada byte armazena uma linha da máscara, um elemento em cada um dos seus oito bits. O primeiro conjunto armazena quais pontos são tanto claros quanto escuros em forma de bits em '1', sendo os demais mantidos em '0'. Já o segundo conjunto armazena quais são claros em forma de bits em '1' e quais são escuros em forma de bits em '0', assumindo os

pontos tanto claros como escuros como sendo claros (bits em '1').

Para executarmos a correlação com máscaras de forma eficiente, armazena-se os os pels da imagem em bytes, cada um contendo um pel armazenado em cada um dos seus 8 bits. Tomam-se 8 desses bytes de modo a termos uma área da imagem de 8 por 8 pels. Executa-se a operação lógica OU entre cada um desses bytes com um byte do conjunto de 8 bytes que guarda os elementos do template que podem ser tanto claros como escuros. O resultado de cada uma dessas operações contém uma imagem onde os elementos que podiam ser tanto claros como escuros foram transformados em claros. Compara-se então esses resultados com o segundo conjunto de 8 bytes e se todos forem iguais, a área da imagem tomada tem as características definidas pela máscara.

```
(INÍCIO)
INICIALIZAR MATRIZ
TANTO_CLARO_QUANTO_ESCURO[8]
INICIALIZAR MATRIZ CLAROS&ESCUROS[8]
INICIALIZAR MATRIZ IMAGEM[8]
DE I=1 ATÉ 8 FAÇA
| IMAGEM[I] = IMAGEM[I] OU
|     TANTO_CLARO_QUANTO_ESCURO[I]
| SE IMAGEM[I] <> CLAROS&ESCUROS[I]
| | NAO COINCIDIU (FIM)
COINCIDIU (FIM)
```

Figura 3: Algoritmo de Correlação com máscaras

### 3.2 Localização de uma região característica.

A imagem a ser processada é lida linha a linha em um conjunto de bytes com tamanho suficiente para abrigar uma linha completa de 1728 pels, sendo colocado 1 pel em cada bit. São mantidas na memória sempre 8 linhas consecutivas da imagem. Toma-se então a primeira word de cada linha e aplica-se sobre o byte mais significativo de todas elas o processo de correlação com máscara para verificar se essa região da imagem de 8 por 8 pels tem as características desejadas.

Se a região da imagem correspondente à máscara não foi encontrada, fazemos um deslocamento lógico de um bit para esquerda nas 8 words que havíamos tomado anteriormente e repetimos o processo num total de 8 vezes. Assim garantimos que testamos a existência do padrão da máscara ponto por ponto nessa área.

Se a região da imagem que corresponde à máscara ainda não foi encontrada, desprezamos o primeiro byte e tomamos a word seguinte após esse byte, repetindo os processos anteriores até o final da linha de imagem ou até encontrarmos a região da imagem procurada.

```
(INÍCIO)
LER 8 LINHAS DA IMAGEM
FAÇA
| FAÇA
| | TOMAR A 1ª WORD DE CADA LINHA
| | DE I=1 ATÉ 8 FAÇA
| | | CORREL. COM MÁSCARA S/ OS 8 BYTES
| | | MAIS SIGNIFICATIVOS DAS 8 WORDS
| | | SE COINCIDIU
| | | | ACHOU (FIM)
| | | DESLOCAR P/ ESQ. 1 BIT NAS 8 WORDS
| | DESPREZAR O 1º BYTE DE CADA LINHA
| ENQUANTO AS LINHAS NAO TERMINAREM
| DESPREZAR A PRIMEIRA LINHA DAS 8
| LER UMA LINHA AO FINAL DAS 7 RESTANTES
ENQUANTO NÃO TERMINAR A IMAGEM
NÃO ACHOU (FIM)
```

Figura 4: Algoritmo para busca de uma região da imagem código.

Se ainda assim não obtivermos sucesso na procura, desprezamos a primeira linha de imagem anteriormente lida, consideramos a segunda como sendo a nova primeira linha e assim sucessivamente até que a oitava linha passe a ser a nova sétima linha. Lemos mais uma linha da imagem para a memória de modo a essa ser a nova oitava linha e repetimos os processos anteriores até o final da imagem ou encontrarmos a região da imagem código procurada.

### 3.3 Localização da Imagem Código.

A procura de primeira região da imagem código é realizada em toda a imagem a ser processada. Porém, ao encontrarmos a primeira região, iniciamos a busca da segunda região em uma área bem definida da imagem, que é a área onde deve estar essa segunda região se estivermos na pista certa da imagem código. Se nessa área limitada não for encontrada a segunda região característica da imagem, significa que a primeira região encontrada foi uma pista falsa, não pertencendo à imagem código desejada.

Encontrada a segunda região, partimos para a busca da terceira, quarta e quantas mais houverem sido definidas para caracterização da imagem código desejada. Cada região seguinte é procurada em uma área bem específica, baseada nas informações colhidas pela localização das regiões anteriores.

O tamanho e a localização das áreas de procura das regiões após a primeira é que definem a tolerância do algoritmo à rotações e à distorções dimensionais na imagem código. Observar que quanto maior a tolerância, maior o tempo de processamento para procurar possíveis ocorrências da imagem código e

maior a suscetibilidade do algoritmo a encontrar uma imagem falsa. Por outro lado, se não fornecermos tolerância suficiente, maiores as chances do algoritmo não encontrar a imagem código.

Observamos também que apesar de ser possível aceitar pequenas rotações na imagem código, estas devem estar limitadas ao ponto que as regiões características não sejam distorcidas (pela rotação) acima do que tolera os elementos tanto claros quanto escuros definidos nas máscaras.

```
(INÍCIO)
PROCURAR A 1ª REGIÃO DA IMAGEM CÓDIGO
SE ACHOU
| PROCURAR A 2ª REGIÃO DA IMAGEM CÓDIGO
| SE ACHOU
| | PROCURAR A 3ª REGIÃO DA IMAGEM CÓD.
| | (buscar quantas regiões tiverem sido definidas)
| | SE ACHOU
| | | LOCALIZADA IMAGEM CÓDIGO. (FIM)
NÃO FOI LOCALIZADA IMAGEM CÓDIGO.(FIM)
```

Figura 5: Algoritmo para localizar a Imagem Código

#### 4. Resultados páticos.

Nos teste realizados com o processo de correlação com máscaras observou-se que o mesmo pode ser bastante sensível a ruído sobre a região a ser identificada. Todavia o processo fac-símile, devido a codificação que a imagem sofre na transmissão, causa distorções bastante características. Essas podem inclusive ser controladas a favor do algoritmo durante a decodificação da imagem minimizando os possíveis problemas.

Pequenas rotações na imagem código não chegam a alterar as regiões características escolhidas para a localização. Apesar da rotação da imagem principal, as regiões características se mantém ortogonais, não prejudicando o processo de correlação com máscaras. Também ocorre que as regiões características são procuradas em uma área em torno do ponto em que elas deveriam estar, não sendo o processo atrapalhado por pequenas rotações na imagem código.

Observou-se que a determinação das máscaras influencia sobremaneira a eficiência do algoritmo. Uma máscara com pouca tolerância (poucos pels indeterminados) faz com que a menor distorção impeça a localização. Já uma máscara muito tolerante causa falsas localizações que aumentam muito o tempo de processamento devido à verificações adicionais infrutíferas.

No caso particular da imagem código apresentada na figura 1, obviamente que rotações múltiplas de 90 graus não modificam em nada a borda a ser localizada.

Conseguimos, também nesse caso particular, tolerar rotações de até sete graus dentro de um compromisso entre velocidade de localização, tolerância à distorções e segurança em não aceitar imagens falsas.

A prática mostrou também que as transmissões fac-símile não apresentam normalmente distorções na imagem. Quando isso acontece, também a mensagem do usuário é danificada e, naturalmente, é solicitada a retransmissão.

Em testes realizados, utilizando uma imagem de uma típica carta comercial contendo a imagem código, conseguiu-se localizar e decodificar o destinatário em menos de sete segundos de processamento em um microcomputador do tipo AT-286, 16 MHz. Esse dado não comprova a eficiência do algoritmo mas demonstra que a implementação prática é factível.

#### 5. Aplicação.

O desenvolvimento do algoritmo de localização de imagens código ora descrito faz parte de um projeto maior de desenvolvimento de um sistema de transmissão e recepção de imagens fac-símile com capacidade de reconhecer o destinatário da mensagem se a mesma contiver a imagem código identificando esse destinatário.

Tenciona-se que o indivíduo que desejar enviar uma mensagem a uma empresa dotada desse sistema de reconhecimento do destinatário sobreponha sobre sua mensagem a imagem código. Isso pode ser conseguido por meio de fotocópia (xerox) ou através de uma folha de rosto já com a imagem código previamente impressa. O remetente da mensagem é incentivado a adotar tal procedimento para que sua mensagem chegue ao destinatário mais rapidamente.

Ao ser recebida essa mensagem que contém a imagem código, o sistema se encarregará de alertar ao usuário sobre a existência de uma mensagem à sua disposição. A mensagem pode ser enviada por rede local ao usuário ou permanecer na memória do equipamento até o momento que o usuário libere a impressão da mesma através de uma senha, tornado a mensagem confidencial.

Em se tratando de uma transmissão entre dois sistemas semelhantes, o próprio sistema se encarrega de apor automaticamente a imagem sobre a mensagem transmitida.

Esse projeto encontra-se em fase final de implementação e é fruto de um acordo de cooperação técnico/científica entre a EQUITEL S.A. e o CEFET PR.

## 6. Conclusões.

Apresentamos um algoritmo prático para localização de uma imagem código dentro de uma página de imagem recebida por fac-símile. Esse algoritmo pode ser usado em outras situações mas deve ser convenientemente adaptado caso a caso. Ele não é infalível mas se combinado com outras técnicas para garantir que a imagem localizada é realmente a desejada ele pode ser muito vantajoso por ser bastante rápido.

O algoritmo será empregado em um projeto maior que poderá ser implementado pela empresa contratante do trabalho e ter utilização real.

## 7. Referências.

Wil J. van Gils, Two-dimensional Dot Codes for Product Identification, IEEE Transactions On Information Theory, Vol. II-33, No.5, september, 1986.

O. Marques, OCR utilizando correlação com máscaras binárias, SIBGRAPI/90

