

# Detecção e Reconhecimento Automático de Placas Veiculares

Verônica P. Nascimento  
Instituto Federal de Educação, Ciência e  
Tecnologia da Bahia  
Rua Emídio dos Santos, Barbalho  
Salvador, Bahia  
veronicapaixao@ifba.edu.br

Antonio C. S. Souza  
Instituto Federal de Educação, Ciência e  
Tecnologia da Bahia  
Rua Emídio dos Santos, Barbalho  
Salvador, Bahia  
antoniocarlos@ifba.edu.br

## RESUMO

Reconhecimento de placas veiculares é desejável em várias aplicações, como monitoramento de estacionamento veicular em locais públicos e privados e verificação da situação do veículo pelas instituições reguladoras de trânsito. Mesmo que existam atualmente várias abordagens propostas para resolver o problema de detecção da placa veicular em uma imagem, poucas são desenvolvidas para reconhecer as placas veiculares de forma automática e interativa sem o uso de sistemas embarcados ou *hardwares* especializados. Neste trabalho, é apresentada uma técnica para detecção e reconhecimento automático de placas veiculares em imagens adquiridas por câmeras comuns. A partir de uma avaliação da abordagem proposta em um *dataset* com 400 imagens, os resultados obtidos mostram que o algoritmo de detecção desenvolvido é genérico o suficiente para detectar placas veiculares de diferentes nacionalidades com taxa de precisão de 80%, e uma taxa de acerto de 60% para reconhecimento de placas veiculares brasileiras.

## Palavras-chave

Detecção, Reconhecimento, Placas Veiculares

## 1. INTRODUÇÃO

Devido ao crescente aumento da frota veicular no mundo, o reconhecimento de placas veiculares se tornou essencial para a engenharia de tráfego, desempenhando um papel importante em várias aplicações, como: controle de acesso a estacionamentos [31], controle de segurança em áreas de acesso restrito [29], monitoramento de tráfego [22], garantia da aplicação das leis de trânsito, sistemas de transporte inteligentes [21], rastreamento de veículos roubados, aplicação automática de multas e coleta de pedágios [27].

Tradicionalmente, as placas veiculares são identificadas da seguinte forma: equipamentos distribuídos em vias públicas e ou locais privados fazem a captura das imagens contendo

os veículos que trafegam na via, as imagens são enviadas para um setor responsável pelo monitoramento do local, que faz a identificação das placas veiculares. O problema desta abordagem é que ela é custosa, manual e tediosa [13].

Os métodos desenvolvidos para o reconhecimento automático de placas veiculares podem ser divididos em três abordagens. A primeira consiste na instalação de um dispositivo no veículo capaz de transmitir um código de identificação (como a própria placa do veículo) para a aplicação. Apesar de essa abordagem garantir altíssima precisão para o reconhecimento da placa veicular, na prática ela não provê alta escalabilidade, pois o dispositivo é instalado em cada veículo a ser monitorado para que a placa seja reconhecida, além disso esta abordagem possui um custo associado à instalação do dispositivo. A segunda abordagem consiste no uso de equipamentos mais especializados, como radares e sensores infra-vermelhos, instalados em uma posição fixa para captura da placa veicular. O uso de sensores garante alta precisão na detecção da placas e reconhecimento dos caracteres da placa, porém estes exigem alto custo para implantação. A abordagem de menor custo consiste no uso de técnicas capazes de realizar o reconhecimento de placas veiculares em imagens capturadas por câmeras convencionais. Nesse caso, o custo associado à aquisição do *software* capaz de realizar esta tarefa seria compensado pela redução do custo obtida com a troca dos sensores especializados por câmeras comuns.

Muitas técnicas já foram desenvolvidas para reconhecimento automático de placas veiculares em imagens comuns, contudo a maioria demanda segundos para realizar o processamento de uma única imagem. O tempo de processamento dessas soluções pode ser impraticável num cenário real, já que as aplicações que utilizam detecção e reconhecimento de placas veiculares, geralmente, exigem uma resposta urgente (poucos milissegundos) para uma maior eficácia. Nestes casos, o ideal seria realizar uma técnica interativa, disponibilizando de forma quase imediata as informações que podem ser obtidas a partir do reconhecimento da placa veicular para outras aplicações.

O principal objetivo deste trabalho é apresentar uma técnica interativa para detecção e reconhecimento automático de placas veiculares em imagens capturadas por câmeras convencionais. A partir do uso de várias técnicas da área de processamento de imagens, é possível realizar a detecção precisa das placas veiculares independente de fatores como



(a) Imagem colorida



(b) Imagem em tons de cinza



(c) Imagem binária

**Figura 1: Tipos de imagens digitais. As placas foram borradas por questões de privacidade.**

localização, iluminação, inclinação e tamanho da placa na imagem. Além disso, o algoritmo desenvolvido neste trabalho para reconhecimento óptico geral de caracteres pode ser utilizado de forma satisfatória para o reconhecimento das placas veiculares brasileiras.

O restante deste trabalho está estruturado da seguinte forma: A fundamentação teórica sobre as áreas de processamento de imagens e reconhecimento de caracteres é descrito na Seção 2. Na Seção 3, uma revisão dos trabalhos recentes desenvolvidos para reconhecimento de placas é apresentada. Na Seção 4, a abordagem proposta neste trabalho é apresentada. Na Seção 5, os resultados são apresentados e discutidos. Por fim, na Seção 6, são feitas as considerações finais sobre o trabalho, bem como sugestões de possíveis trabalhos futuros.



(a) Imagem colorida



(b) Imagem binária  
(limiar = 64)



(c) Imagem binária  
(Otsu)

**Figura 2: Binarização produzida pelo uso de limiar fixo (b) e limiar calculado pelo método de Otsu [25] (c). As placas foram borradas por questões de privacidade.**

## 2. FUNDAMENTAÇÃO TEÓRICA

### 2.1 Processamento de Imagens Digitais

Uma imagem digital pode ser definida como uma matriz cujos elementos identificam um ponto (ou *pixel*) na imagem. Tipicamente, cada *pixel* é associado com um valor inteiro, que varia de 0 a 255, correspondente a uma intensidade de cor da imagem. Nesse caso, 0 indica a ausência de cor, enquanto 255 indica intensidade máxima de cor. A definição de cor varia de acordo com o tipo de imagem digital utilizado. De forma geral, uma imagem digital pode ser classificada em três tipos [12] (Figura 1):

- **Imagem colorida:** aquela em cada *pixel* pode possuir

um valor no intervalo entre 0 e 255 para cada canal de cor da imagem. Esse tipo de imagem possui três canais de cor: vermelho, verde e azul. A combinação desses três canais de cor básicos é que define a aparência colorida da imagem (Figura 1-(a)).

- **Imagem em tons de cinza:** aquela em cada *pixel* pode possuir apenas um valor no intervalo entre 0 e 255. Assim como o nome do tipo da imagem já indica, esse tipo de imagem possui variações de cinza em sua aparência (Figura 1-(b));
- **Imagem binária:** aquela em que cada *pixel* pode possuir apenas um dos dois valores: 0 ou 255. Dessa forma, a imagem apenas possui *pixels* de cor preta ou *pixels* de cor branca (Figura 1-(c));

Como as imagens binárias e em tons de cinza possuem apenas um único canal de cor, são mais fáceis e rápidas de serem processadas do que as imagens coloridas. Uma imagem colorida pode ser facilmente convertida em tons de cinza a partir da média aritmética ou ponderada das intensidades de cor vermelha, verde e azul, resultando em uma única tonalidade cinza. Já uma imagem em tons de cinza pode ser convertida para uma imagem binária de várias formas. A mais simples consiste no uso de um limiar (um valor no intervalo 0 a 255) que atribui a um *pixel* a cor branca se a intensidade em tons de cinza desse *pixel* estiver acima desse limiar, enquanto a cor preta é atribuída caso contrário. Este limiar pode ser definido manualmente (Figura 2-(b)) ou automaticamente a partir de algoritmos como o de Otsu [25], que determina o limiar de acordo com um cálculo de variância sobre a imagem (Figura 2-(c)).

Enquanto a conversão da imagem colorida para a imagem em tons de cinza ou binária é possível, o inverso não é possível, uma vez que a informação de cor no padrão vermelho, verde e azul se perde no processo de conversão para um único canal de cor.

As câmeras digitais produzidas nos dias de hoje proveem captura de imagens coloridas de alta resolução. Ainda assim, técnicas de processamento de imagens podem ser utilizadas para a melhoria da informação visual já contida na imagem [12]. Um exemplo disso está nas técnicas de detecção de bordas. Técnicas desse tipo são muito úteis pois, uma vez com as arestas da imagem detectadas, torna-se possível a realização de operações como: realce das características da imagem, renderização não-fotorrealística da imagem e segmentação de objetos de interesse na imagem (Figura 3).

Os métodos mais comuns para detecção de bordas são os algoritmos de Sobel [33] e Canny [6] (Figura 4). O algoritmo de Sobel utiliza dois *kernels*  $K_x$ ,  $K_y$   $3 \times 3$

$$K_x = \begin{pmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{pmatrix}$$

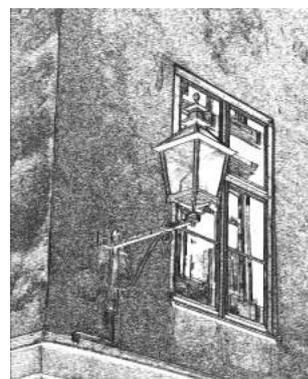
$$K_y = \begin{pmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{pmatrix}$$



(a) Imagem original



(b) Realce de características



(c) Renderização não-fotorrealística

**Figura 3: Operações de processamento de imagens baseadas no uso de detecção de bordas. Imagens retiradas de [10].**

que são aplicados sobre a imagem com o objetivo de detectar as bordas aproximadas nos eixos horizontal e vertical para cada *pixel* da imagem. Esse algoritmo é bastante útil por ser veloz, contudo é susceptível à geração de ruídos (Figura 4-(a)).

O algoritmo de Canny utiliza um esquema mais robusto de detecção de bordas que consiste nos seguintes passos:



(a) Imagem colorida



(b) Bordas detectadas (Sobel)



(c) Bordas detectadas (Canny)

**Figura 4: Bordas detectadas pelos algoritmos de Sobel [33] (b) e Canny [6] (c). As placas foram borradas por questões de privacidade.**

1. Eliminação de ruídos da imagem: Nesse passo, um filtro Gaussiano de borramento é aplicado sobre a imagem original de forma a borrar a imagem e remover os ruídos provenientes da aquisição da imagem;
2. Cálculo dos gradientes da imagem: Uma vez que os ruídos tenham sido removidos ou minimizados, um *kernel* é utilizado para detectar as bordas da imagem. O próprio *kernel* do Sobel pode ser utilizado nessa etapa;
3. Supressão não-máxima da imagem: *Pixels* não considerados parte da borda da imagem são eliminados nesta etapa do algoritmo;
4. Rastreamento de bordas por histerese: Uma análise da vizinhança  $3 \times 3$  é feita para determinar se uma

borda está ou não conectada a uma borda vizinha da imagem;

O algoritmo de Canny é um pouco mais lento do que o algoritmo de Sobel, porém oferece detecção de borda com maior precisão (Figura 4-(b)).

Uma vez com as bordas ou arestas de uma imagem detectadas, torna-se possível o uso de um algoritmo de segmentação para extração de regiões de interesse da imagem. Um dos métodos mais simples para segmentação de regiões de interesse se baseia na ideia de detecção de contornos [34]. Um contorno pode ser simplesmente definido como uma lista de pontos que representam uma curva (ou reta) em uma imagem, sendo que cada elemento dessa lista armazena informação sobre a localização do próximo ponto dentro da curva [5].

Nesse método, as bordas geradas pelos métodos de Sobel ou Canny são rastreadas de forma a se encontrar contornos ou polígonos fechados que contém um objeto ou região de interesse.

A área de processamento de imagens contém muitas de técnicas disponíveis para a realização de inúmeras tarefas [12]. Contudo, nessa seção o foco foi dado principalmente para os algoritmos de conversão entre tipos de imagens, detecção de bordas e segmentação de objetos de interesse pois estes são os mais utilizados para detecção e reconhecimento de placas veiculares [7].

## 2.2 Reconhecimento de Placas Veiculares

De acordo com [7], o reconhecimento de placas veiculares pode ser feito a partir de quatro etapas sequenciais (Figura 5)

- A aquisição da imagem com a placa veicular;
- O processamento da imagem para detecção da placa veicular;
- A extração/segmentação da placa veicular da imagem;
- O reconhecimento dos caracteres contidos na placa veicular;

Assim como visto na Seção 1, a aquisição da imagem com a placa veicular pode ser feita de várias formas: a partir de câmeras (Figura 5-(a)), sensores infra-vermelhos, radares, entre outros. A forma como a imagem é adquirida afeta de forma considerável as outras etapas do reconhecimento de placas veiculares.

Uma vez com a imagem adquirida, o próximo passo consiste na detecção da placa veicular na imagem (Figura 5-(b)). Uma placa veicular pode estar localizada em qualquer lugar da imagem [7]. Contudo, ao invés de processar cada *pixel* da imagem em busca dessa placa veicular, uma forma de otimizar esse processo consiste em distinguir a placa veicular pelas suas características e então processar apenas aqueles *pixels* que possuem essas características. Exemplos de características relevantes para localização de placas veiculares



(a) Imagem original



(b) Placa detectada



(c) Placa segmentada

OLG-4332

(d) Placa reconhecida

**Figura 5: Visão geral do processo de reconhecimento de placas veiculares: dada uma imagem com uma placa veicular (a), a placa é detectada (b), segmentada (c) e enfim os caracteres são reconhecidos (d) pelo sistema de reconhecimento de placas veiculares. As placas foram borradas por questões de privacidade.**

são o seu formato retangular e a existência de caracteres dentro da placa. Ambos podem ser facilmente detectados a partir de métodos de detecção de bordas.

Dada a localização da placa detectada na imagem, a região isolada da placa veicular deve ser então segmentada (Figura 5-(c)) para a próxima etapa de reconhecimento dos caracteres contido na placa.

A etapa final do reconhecimento da placa veicular consiste no reconhecimento de cada um dos caracteres contidos na imagem na localização da placa veicular (Figura 5-(d)). Nesse caso, algoritmos de correspondência de padrões (*template matching*) ou algoritmos baseados em aprendizagem de máquina são utilizados para a detecção dos caracteres.

### 3. TRABALHOS RELACIONADOS

Muitos métodos têm sido propostos na literatura para o reconhecimento automático de placas veiculares. Nesta seção, os trabalhos mais relevantes são classificados de acordo com o tipo de abordagem utilizado para o reconhecimento das placas veiculares. Uma revisão mais completa dos trabalhos relacionados está além do escopo desta monografia. Esta revisão pode ser encontrada em [7].

#### 3.1 Detecção de Placas Veiculares

##### 3.1.1 Detecção por Localização de Bordas

Uma vez que a placa veicular tem um formato retangular característico, métodos de detecção de bordas são comumente utilizados para localizar essas retângulos e então detectar a placa veicular [36].

Algoritmos tradicionais como o algoritmo de Sobel [17, 28] e a transformada de Hough para detecção de linhas [16, 8] são os métodos mais comuns para detecção de placas veiculares por localização de bordas. Contudo, outras abordagens específicas da área de detecção de placas veiculares como detecção vertical [4] e detecção baseada em blocos [20] também foram propostas.

Enquanto os métodos baseados em localização de bordas são rápidos e eficientes, eles exigem a detecção de bordas precisas, sem ruídos, para permitir a detecção das placas veiculares com alta qualidade [7].

##### 3.1.2 Detecção pelas Características de Cor

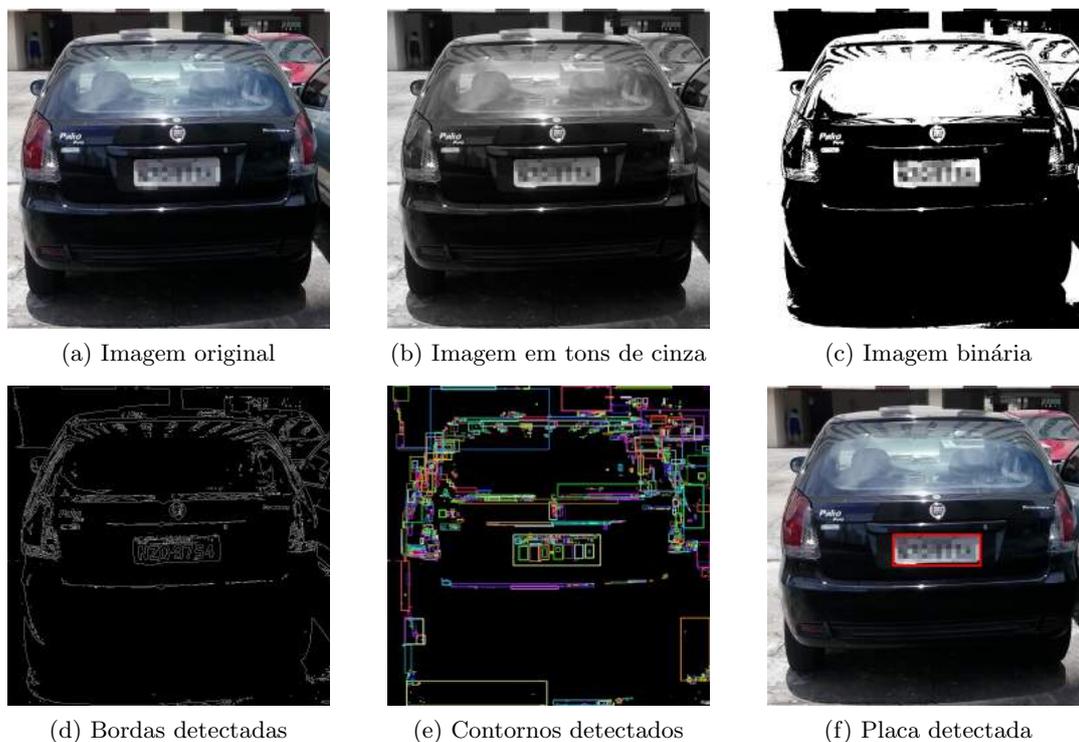
Técnicas desta categoria utilizam uma combinação de cor da placa e dos caracteres contidos na placa de forma a detectar a região com a placa veicular na imagem [30]. Técnicas baseadas em redes neurais [19] ou algoritmos genéticos [35] são utilizadas para detectar padrões de cor das placas na imagem.

A detecção de placas veiculares a partir da localização das características de cor da imagem tem a vantagem de detectar placas deformadas e inclinadas. Contudo, processamento de cor na imagem é muito sensível à ruídos e às condições de iluminação da imagem capturada, o que reduz a precisão das técnicas dessa categoria.

##### 3.1.3 Detecção por Localização de Caracteres

Uma forma alternativa de detecção de placas veiculares consiste na localização dos caracteres contidos na placa. No momento em que os caracteres são encontrados, a região da placa é localizada. Apesar de haver uma série de formas de se reconhecer caracteres em uma imagem, assim como descrito na próxima subseção, as abordagens que utilizam a localização de caracteres para detectar placas geralmente utilizam algoritmos de aprendizagem de máquina como redes neurais [24] e máquina de vetor suporte [14], devido a boa precisão dada por essas abordagens.

Esse tipo de abordagem possui três problemas principais: 1. o processo de localizar caracteres em uma imagem demanda alto tempo de processamento, 2. a precisão dessa abordagem é afetada no caso da existência de texto em outras regiões da imagem fora da placa, 3. a placa não pode ser localizada



**Figura 6:** Visão geral do processo de detecção automática de placas veiculares proposto neste trabalho: dada uma imagem com uma placa veicular (a), esta é convertida para tons de cinza (b) e binarizada (c). Em seguida, as bordas da imagem binária são detectadas (d) e estas bordas são agrupadas de acordo com seus contornos (e). Dentre todos os contornos detectados na fase anterior, uma série de critérios são utilizados para definir qual desses contornos representa a placa veicular presente na cena (f). As placas foram borradas por questões de privacidade.

caso os caracteres dentro dela não sejam detectados, devido às condições de ruído e iluminação da própria imagem.

## 3.2 Reconhecimento de Caracteres

### 3.2.1 Reconhecimento por Correspondência de Padrão

O método mais simples para reconhecimento de caracteres em uma imagem se dá a partir da correspondência de padrão [26]. Uma vez que cada caractere tenha sido segmentado dentro da placa veicular, cada imagem de uma base de dados de caracteres é comparada aos caracteres localizados na imagem da placa veicular. A similaridade entre ambas as imagens é medida e o caractere contido na melhor imagem da base de dados é escolhido como o caractere localizado da placa.

Apesar de ser o método mais simples e fácil para reconhecimento de caracteres, se um caractere da placa tem fonte ou tamanho diferente dos caracteres contidos na base de dados, o método de correspondência de padrões produz resultados incorretos. Ruídos e condições de iluminação afetam a precisão do reconhecimento das placas.

### 3.2.2 Reconhecimento por Extração de Características

Uma forma alternativa, mais robusta, para reconhecimento de caracteres consiste na extração e comparação de características extraídas por cada caractere. Um vetor de características é extraído para cada caractere da placa e é comparado com um vetor de características pré-calculado, de forma que a similaridade entre os vetores de características determina qual o caractere da placa veicular.

O vetor de características pode ser gerado a partir da divisão do caractere de uma imagem binária em blocos  $3 \times 3$  de pixels, de forma que o número de pixels pretos são contados em cada bloco [3]. Uma forma alternativa consiste em representar o caractere a partir da detecção do seus contornos [15, 2]. Após a extração de características, classificadores treinados podem ser utilizados para o reconhecimento dos caracteres [18, 23].

O uso de vetor de características para reconhecimento de caracteres tem elevada precisão, sendo robusto a eventuais distorções na imagem. Contudo, a extração do vetor de características na imagem e o uso desse vetor para reconhecimento de caracteres consome mais tempo do que a abordagem por correspondência de padrão.

## 4. RECONHECIMENTO AUTOMÁTICO DE PLACAS VEICULARES

Nesta seção é apresentado o algoritmo desenvolvido para reconhecimento automático de uma placa veicular. Dada uma imagem que contém uma placa veicular, esta é primeiro localizada na cena a partir do uso de algoritmos de detecção de bordas e contornos e em seguida, os caracteres são reconhecidos a partir do uso de um classificador treinado para o reconhecimento de caracteres. Uma visão geral do algoritmo proposto pode ser visto na Figura 6. Cada um dos passos desse algoritmo é explicado em detalhes nas subseções abaixo.

### 4.1 Detecção das Placas Veiculares

O processo de detecção automática de uma placa veicular pode ser descrito nos seguintes passos:

#### 4.1.1 Conversão da imagem colorida para tons de cinza

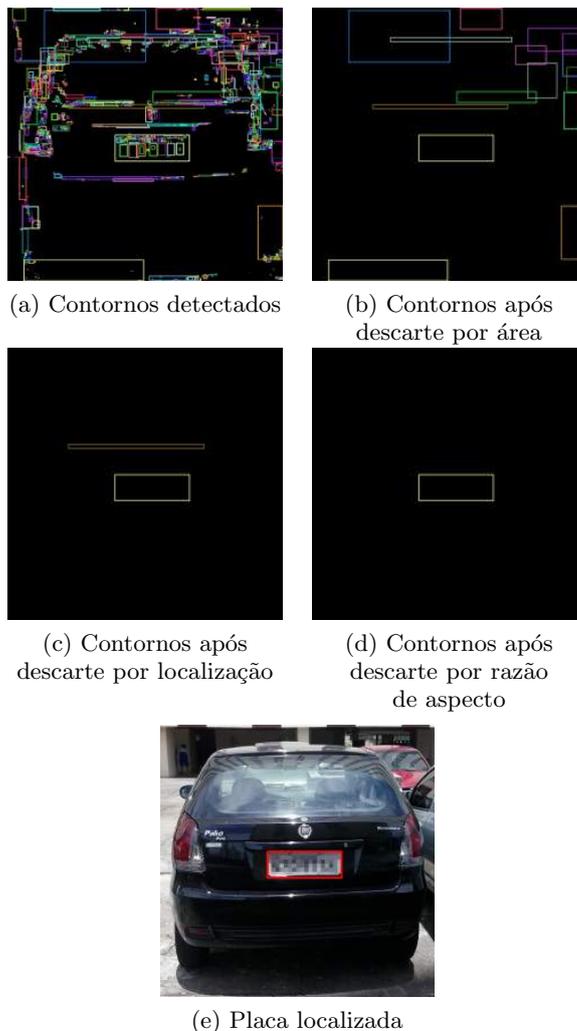
Para a detecção automática de uma placa veicular, primeiro faz-se necessário converter a imagem colorida capturada pela câmera (Figura 6-(a)) para uma imagem em tons de cinza (Figura 6-(b)) a partir do processo de média aritmética das intensidades verde, vermelho e azul de cada cor. Uma das grandes vantagens deste passo é a redução, em três vezes, na quantidade de canais de cor que serão processados nas próximas etapas do algoritmo para a detecção da placa veicular;

#### 4.1.2 Conversão da imagem em tons de cinza para binária

Um dos fatores que caracterizam uma placa veicular em uma imagem é o fato de a placa veicular possuir cor majoritariamente branca. Contudo, em uma imagem em tons de cinza, a placa não necessariamente possuirá cor branca, mas sim uma cor levemente acinzentada por conta de eventuais sombras lançadas sobre a placa, ou mesmo pelo processo de captura da imagem. Dessa forma, é feita a binarização da imagem em tons de cinza, de forma que a placa, por estar presente em uma região mais clara da imagem, ficará com a cor branca (com exceção aos seus caracteres, que possuirão cor preta) (Figura 6-(c)). O algoritmo de Otsu [25] é aplicado nesta etapa pois provê binarização automática, e de alta qualidade, sobre a imagem em tons de cinza (Seção 2);

#### 4.1.3 Detecção de bordas na imagem binária

Apesar da conversão da imagem em tons de cinza para a imagem binária prover uma distinção da região que contém a placa em relação às demais na imagem (Figura 6-(c)), também é atribuída cor branca para outras regiões da imagem, como o asfalto na rua, o vidro do carro, etc. Contudo, diferentemente dessas regiões, a placa veicular possui um formato retangular e uma proporção entre sua largura e altura que facilmente a distinguem na imagem. De forma a representar a placa não mais de acordo com a sua cor, mas sim a partir das características de dimensão, é utilizado um algoritmo de detecção de bordas para localizar todas as regiões de arestas localizadas na imagem binária. Nessa imagem com as bordas detectadas, a placa é representada apenas pelas bordas que ela contém (Figura 6-(d)). O algoritmo de



**Figura 7: Processo de detecção da placa veicular a partir da remoção dos retângulos que não pertencem a uma placa veicular. Critérios como tamanho (b), localização (c) e razão de aspecto (d) são utilizados para definir qual dos retângulos (a) consiste numa placa veicular (e). As placas foram borradas por questões de privacidade.**

Canny [6] é utilizado para detectar as bordas devido à sua robustez à presença de ruídos e qualidade superior em relação às outras abordagens de detecção de bordas existentes (Seção 2).

#### 4.1.4 Agrupamento de contornos sobre as bordas detectadas

Enquanto o algoritmo de Canny provê a detecção de bordas sobre a imagem binária, cada *pixel* apenas tem conhecimento se ele está contido em uma região de borda. Para a detecção da região com a placa veicular, é necessário conectar todos os *pixels* vizinhos que estão localizados sobre uma mesma região de borda (Figura 6-(e)). Nesse caso, o algoritmo de detecção de contornos proposto em [34] é utilizado. Este algoritmo faz uma varredura da imagem e conecta os *pixels*

vizinhos que estão localizados em cima da borda da imagem. Apenas são considerados contornos da imagem aqueles grupos de *pixels* que formam regiões retangulares na imagem.

#### 4.1.5 Detecção do contorno com a placa veicular

Assim como pode ser visto na Figura 6-(e), o passo anterior detecta inúmeras regiões retangulares sobre as bordas detectadas na Figura 6-(d). Nesse caso, uma série de critérios são utilizados para descartar da imagem aqueles contornos que não representam uma região de placa veicular. Estes critérios foram definidos de acordo com uma avaliação prévia feita num *dataset* com placas veiculares. Os critérios definidos foram:

- A área de uma placa veicular ocupa mais que 0.5% da área total da imagem - Este critério faz sentido não somente porque elimina muitos retângulos pequenos que foram detectados no passo anterior, como também aquelas placas que estão tão distantes da câmera, que não seria possível realizar o reconhecimento dos caracteres da placa (Figura 7-(b));
- A área de uma placa veicular ocupa menos que 10% da área total da imagem - Esse critério permite a eliminação de retângulos grandes tipicamente detectados nos vidros traseiros dos carros (Figura 7-(b));
- Uma placa veicular geralmente não está posicionada nos extremos da imagem - Neste trabalho, não foram consideradas placas veiculares parcialmente presentes na imagem, pois isso prejudicaria o reconhecimento dos caracteres contidos na placa veicular. Dessa forma, as placas geralmente estão localizadas no centro da imagem. Na prática, são descartados todos os retângulos cujas extremidades horizontais e verticais estão localizadas fora da faixa de 20% a 80% da largura e altura total da imagem (Figura 7-(c));
- A razão de proporção do tamanho de uma placa veicular (divisão entre a largura e a altura da placa), seja ela de qual nacionalidade for, está entre 2.5 e 5 unidades - Quadrados ou retângulos cuja razão de proporção não correspondem a uma placa veicular são descartados com esse critério (Figura 7-(d));
- Caso os critérios de seleção definidos anteriormente mantenham vários retângulos na imagem, assume-se que o retângulo representante da placa veicular será aquele localizado na posição mais central da imagem. Se ainda assim houverem vários retângulos localizados na região central da imagem, assume-se que a placa está localizada na posição do retângulo com maior área - Apesar da natureza relativamente empírica desta abordagem, ela se provou bastante útil na prática como critério de 'desempate' para a seleção do retângulo com a placa veicular;

O resultado dessa etapa é localização da placa veicular (Figura 6-(f)). Na prática, apenas os pontos dos cantos superior-direito e inferior-esquerdo do retângulo são armazenados e enviados para a próxima etapa de reconhecimento dos caracteres.



(a) Placas segmentadas incorretamente



(b) Placas segmentadas corretamente

**Figura 8: Placas segmentadas de forma incorreta devido à presença de caracteres adicionais, como sigla do estado e cidade de registro da placa, ou por conta do superdimensionamento da placa (a). A partir de alguns ajustes, é possível segmentar apenas a região com os caracteres contidos na placa (b). As placas foram borradas por questões de privacidade.**

## 4.2 Reconhecimento de Caracteres

Com o retângulo que contendo a localização da placa veicular na imagem (Figura 5-(b)), é feita a segmentação da placa veicular (Figura 5-(c)) e em seguida é utilizado um algoritmo de reconhecimento de caracteres para reconhecimento automático da placa veicular (Figura 5-(d)). Todo este processo é descrito em mais detalhes à seguir.

### 4.2.1 Segmentação da placa veicular

Para a segmentação da placa veicular na imagem, inicialmente cria-se uma nova imagem cuja dimensão é a mesma do retângulo detectado no passo anterior. Esta imagem recebe apenas os *pixels* contidos dentro da região estimada da placa veicular. Contudo, a placa segmentada tipicamente contém informações que vão além dos caracteres alfanuméricos que identificam a placa veicular. No Brasil, por exemplo, além dos sete caracteres alfanuméricos (um conjunto de três letras e quatro números) que identificam a placa veicular, a sigla do estado e o nome da cidade onde a placa foi registrada também estão contidos dentro da imagem da placa veicular (Figura 8-(a)). Além disso, o algoritmo de detecção pode gerar um retângulo com dimensão lateral maior do que a região com a placa veicular, o que pode prejudicar o algoritmo para reconhecimento de caracteres (Figura 8-(a)). Em ambos os casos, é necessário realizar um ajuste para restringir a região da placa detectada para a região que contém apenas os caracteres alfanuméricos da placa (Figura 8-(b)). Dessa forma, é possível não somente reduzir o tempo



**Figura 9:** Diferentes tipos de placas veiculares brasileiras. As placas se diferenciam em termos de fontes dos caracteres, bem como cor de fundo da placa e cor de cada caractere. As placas foram borradas por questões de privacidade.

de processamento necessário para reconhecimento da placa veicular, uma vez que uma imagem de tamanho menor será utilizada para representar a placa, como também será possível aumentar a precisão do algoritmo de reconhecimento automático da placa veicular.

O procedimento para melhoria da segmentação da placa veicular consiste no seguinte: uma vez que se sabe que a localização das informações 'adicionais' à placa veicular é logo acima da localização dos caracteres alfanuméricos, segmenta-se as primeiras 20 linhas da imagem. Para resolver o problema de superdimensionamento lateral na localização das placas veiculares, são segmentadas 60 colunas nos lados direito e esquerdo da imagem. Esse valores foram definidos de forma empírica, mas na prática geram bons resultados, assim como visto na Figura 8.

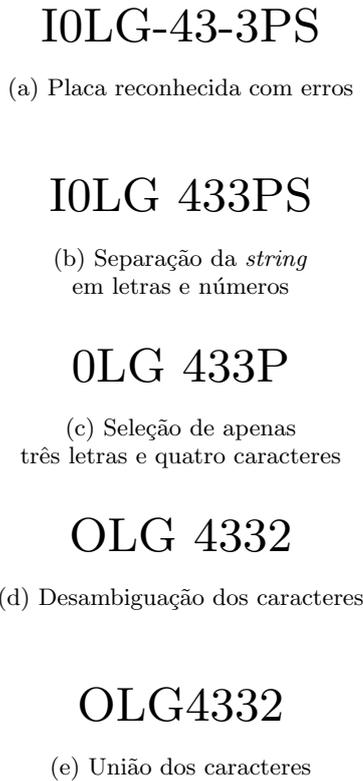


**Figura 10:** Diferentes condições de iluminação obtidas durante a captura das placas veiculares. Sombras, imagens com tonalidade azulada ou escurecida podem prejudicar o reconhecimento dos caracteres na placa. As placas foram borradas por questões de privacidade.

#### 4.2.2 Reconhecimento de cada caractere

Uma vez com a imagem apropriadamente segmentada, é realizado o reconhecimento dos caracteres da placa. Diferentemente dos tipos de fontes encontrados nos textos convencionais, as placas veiculares, mesmo as brasileiras, não possuem um tipo de fonte padrão. Assim como pode ser visto na Figura 9, não somente o estilo da fonte mas a cor dos caracteres pode variar entre placas veiculares, de acordo com o tipo de veículo utilizado. Dessa forma, o algoritmo para reconhecimento de caracteres deve ser robusto o suficiente para lidar com todas essas variações para reconhecer corretamente os caracteres da placa. Para tanto, foi utilizado um algoritmo de reconhecimento de caracteres *open source* já treinado especificamente com várias imagens de placas veiculares segmentadas. Este é aplicado sobre a imagem para o reconhecimento dos caracteres alfanuméricos da placa.

Um dos problemas do algoritmo de reconhecimento de caracteres é que ele é susceptível a ruídos (Figura 10). Ruídos nesse caso estão relacionados não somente às condições da imagem (iluminação, qualidade da câmera), mas também à qualidade da segmentação, assim como visto na subseção anterior. Dessa forma, o algoritmo retorna uma *string* com uma série de imperfeições (espaços em branco, hifens em excesso, caracteres com pontuações, mais de sete caracteres, caracteres reconhecidos incorretamente, entre outros), que devem ser processados para retornar apenas sete caracteres, com três letras e quatro números. O primeiro ajuste feito sobre o algoritmo é a inclusão da lista dos possíveis caracteres



**Figura 11: Processamento dos caracteres reconhecidos. Dada uma placa reconhecida incorretamente (a), o hífen é utilizado para separar as três letras dos quatro números da placa veicular (b). Caracteres adicionais são eliminados (c) e é feita a desambiguação dos caracteres reconhecidos (d). Por fim, é feita a união dos caracteres para representação da placa veicular (e).**

que podem ser reconhecidos pelo algoritmo. Para tanto, foi utilizada a lista '0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ-'. Dessa forma, o algoritmo poderia reconhecer apenas números num intervalo de 0-9, letras no intervalo de A-Z, e o hífen para a placa veicular. A separação entre letras e números é identificada na *string* a partir da detecção do hífen contido na placa veicular. Caso mais de três letras sejam identificadas na placa, arbitrariamente se escolhem as três últimas, uma vez que o canto esquerdo da placa em alguns momentos é identificado como o caractere 'I'. Caso mais de quatro números sejam identificados na placa, escolhe-se apenas as quatro primeiras, pois o canto direito pode ser identificado como o número '1'.

Uma vez com os caracteres das letras e dos números já reconhecidos, faz-se um novo processamento para remover as inconsistências das letras que foram reconhecidos como números e números que forma reconhecidos como letras. O algoritmo de reconhecimento de caracteres tipicamente confunde as letras pelos números (e vice-versa) contidos na Tabela 1. Como a *string* de letras deve conter apenas letras, e a *string* de números deve conter apenas números, é feita a substituição das letras e números reconhecidos incorreta-

Número	Letra
0	D, O
1	I, L
2	B, P
4	F
5	S, Z
6	G
7	Z
8	H, B

**Tabela 1: Letras e números tipicamente confundidos por algoritmos de reconhecimento de caracteres e que devem ser tratados posteriormente pelo algoritmo.**

Item	Configuração
Processador	Intel Core i7
Memória	8GB RAM
Placa de Vídeo	NVIDIA GeForce GTX 660
Sistema Operacional	Windows 7

**Tabela 2: Configuração experimental do computador utilizado para avaliação da abordagem proposta.**

mente, seguindo a Tabela 1.

Por fim, as *strings* das letras e dos números são concatenadas, resultado num único conjunto de caracteres representando a placa veicular.

Por conta dos ruídos e das diferenças de fontes entre as placas veiculares, é possível que o algoritmo de reconhecimento detecte menos de sete caracteres para a placa veicular. Nesse caso, o processamento é feito da mesma forma e o algoritmo retorna a *string*, mesmo com menos de sete caracteres.

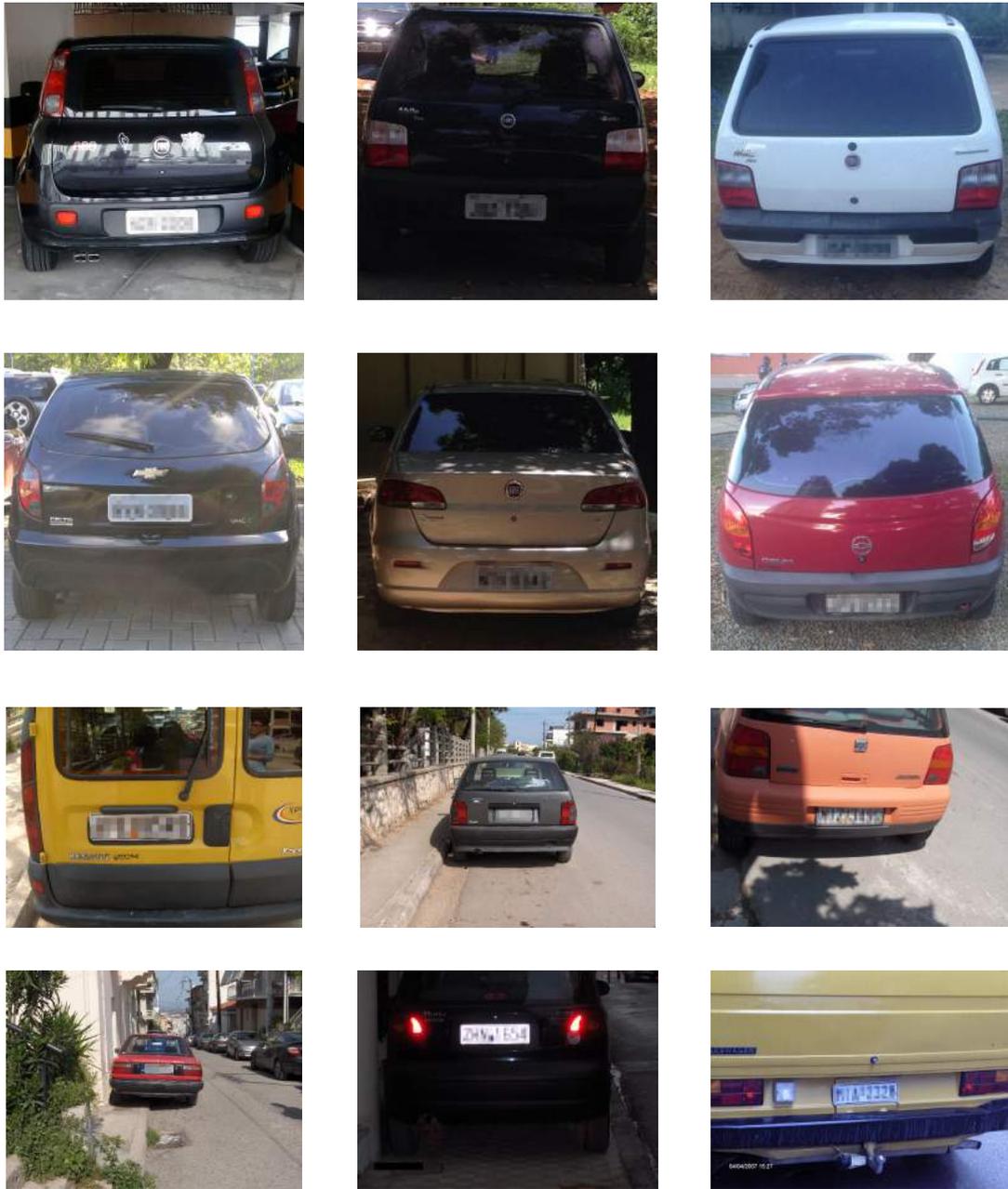
## 5. RESULTADOS E DISCUSSÃO

Nesta seção, é apresentado o ambiente experimental de desenvolvimento e teste da abordagem proposta. Em seguida, a técnica desenvolvida é avaliada em termos de tempo de execução e precisão.

### 5.1 Configuração Experimental

O algoritmo de detecção de placas veiculares foi desenvolvido a partir da biblioteca *OpenCV* [5] em C++, enquanto o algoritmo de reconhecimento de caracteres foi desenvolvido a partir da biblioteca *Tesseract* [32] também em C++. A abordagem proposta foi avaliada em termos de tempo de processamento e precisão a partir de um computador com as configurações mostradas na Tabela 2.

Diferentemente de outras sub-áreas de visão computacional e reconhecimento de padrões, ainda não há um *dataset* padrão com imagens de alta resolução para a avaliação dos algoritmos de detecção e reconhecimento de placas veiculares, apesar dos recentes esforços para a inversão desse cenário [11]. Diferente de outros objetos, uma placa veicular contém informações privadas valiosas que permitem a própria identificação do veículo. Então, por questões de segurança, evita-se a disponibilização do *dataset*, por conta da falta de consentimento daqueles que tiveram suas placas veiculares capturadas pela câmera e populadas no *dataset*. Dessa



**Figura 12:** Algumas imagens do *dataset* utilizado. Imagens variam em condições de iluminação, nacionalidade das placas e distância da câmera. As placas foram borradas por questões de privacidade.

forma, a avaliação da abordagem proposta é feita a partir de um *dataset* montado com imagens de placas veiculares brasileiras capturadas pelos autores deste trabalho e em trabalhos relacionados [9], bem como algumas imagens de placas veiculares internacionais já disponíveis na internet [1]. O *dataset* final de avaliação consiste em 400 imagens, sendo que os autores deste trabalho fizeram a anotação manual da localização e dos caracteres contidos na placa veicular de cada uma das imagens desse *dataset*. Assim como pode ser visto na Figura 12, o *dataset* foi montado de forma a incluir uma variedade de placas veiculares de diferentes nacionalidades, em diferentes condições de iluminação, e capturadas

a distâncias e ângulos variáveis em relação à câmera.

## 5.2 Precisão

O algoritmo de detecção de placas veiculares obteve uma taxa de acerto de 80% sobre o *dataset* de 400 imagens. Além disso, houve uma taxa de apenas 20% na detecção de falsos positivos no *dataset*. Garante-se, dessa forma, que o algoritmo desenvolvido possui boa precisão, avaliada em um *dataset* que agrega diferentes condições que dificultam a localização da placa veicular. Algumas dos resultados do algoritmo de localização de placas veiculares podem ser visto na Figura 13. Contudo, ainda assim, fatores fundamentais para

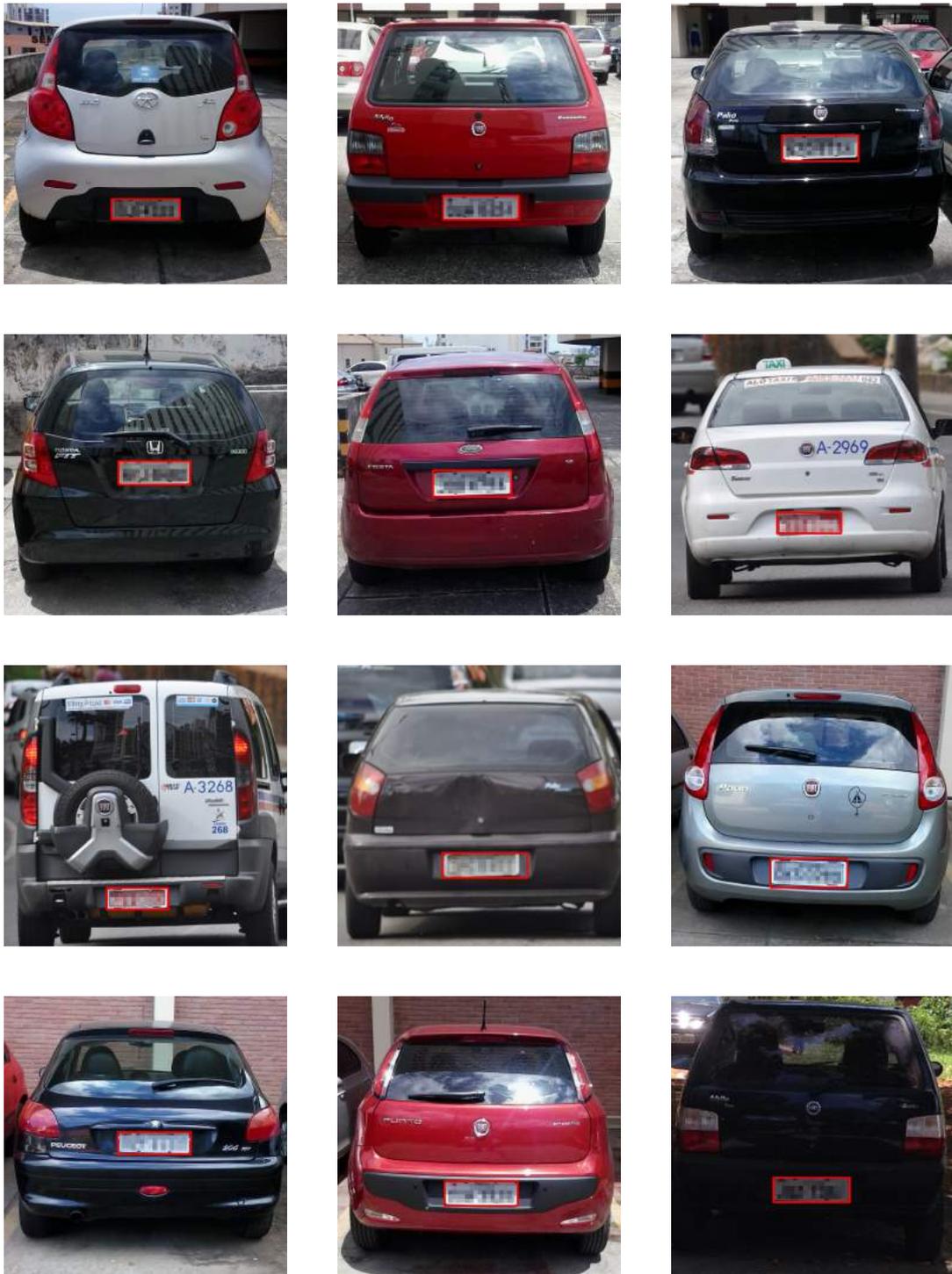


Figura 13: Algumas das imagens utilizadas no *dataset* que foram devidamente localizadas pelo algoritmo proposto. As placas foram borradas por questões de privacidade.

a detecção de falsos positivos como iluminação e a presença de outras regiões retangulares na imagem podem prejudicar a localização da placa veicular (Figura 14).

Em comparação ao trabalho proposto em [9], que também realiza localização de placas veiculares, foi obtido uma pre-

cisão de 80% sobre o *dataset* de 100 imagens utilizado em [9], com a mesma taxa de 20% na detecção de falsos positivos. Enquanto [9] reporta uma precisão de 96%, maior que a obtida no algoritmo proposto neste trabalho, também é reportada uma alta taxa de falsos positivos de 63%. Dessa forma, a abordagem aqui proposta localiza menos placas cor-

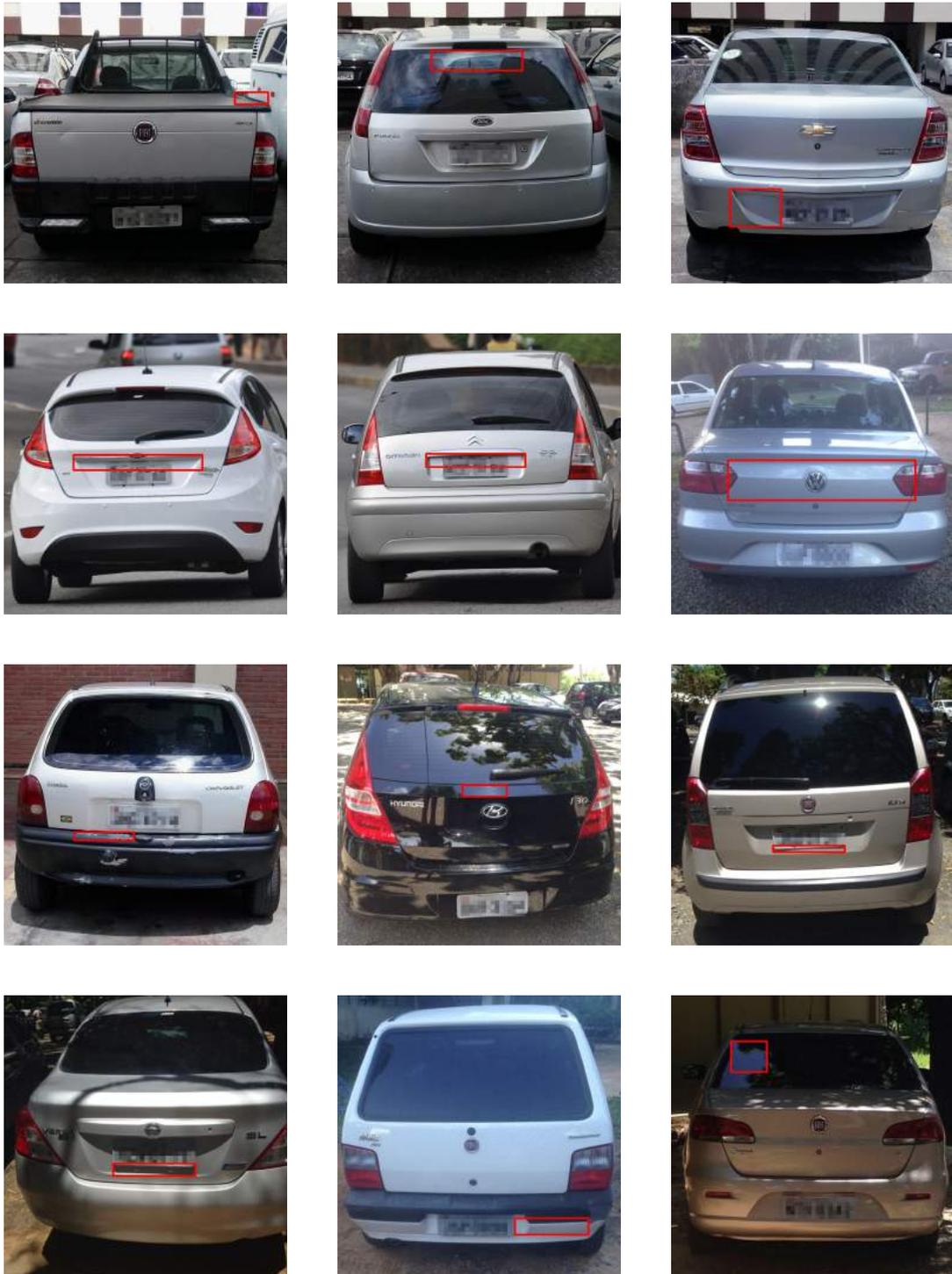


Figura 14: Algumas das imagens utilizadas no *dataset* que não foram devidamente localizadas pelo algoritmo proposto. As placas foram borradas por questões de privacidade.

retas do que [9], contudo possui uma taxa de acerto, quando na localização da placa, 3 vezes maior do que [9].

No que diz respeito ao algoritmo de reconhecimento de caracteres, uma vez que cada nacionalidade possui uma placa veicular com uma fonte e tamanho diferente (Figura 15), foi

optado pelo teste do algoritmo de reconhecimento de caracteres apenas em placas veiculares brasileiras, já que poucos trabalhos na literatura dão atenção ao tipo de placa veicular utilizado no Brasil. Nesse sentido, para essa etapa de reconhecimento de caracteres, apenas o *dataset* de 100 placas veiculares brasileiras de [9] foi utilizado para validação.



Figura 15: As placas produzidas no Brasil (as três primeiras, de cima para baixo) diferentes das placas estrangeiras em termos de fonte, tamanho e conteúdo. As placas foram borradas por questões de privacidade.

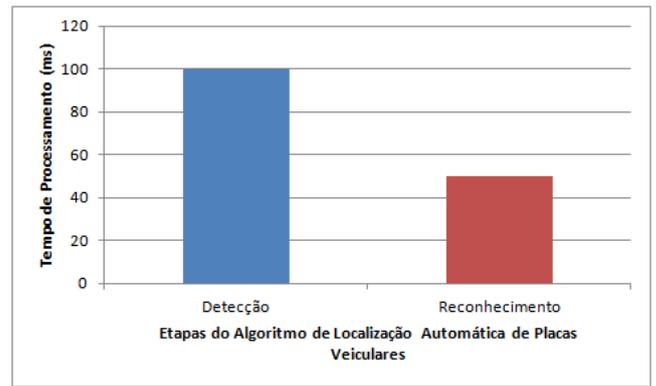


Figura 16: Tempo de processamento obtido (em milissegundos) para as etapas de detecção e reconhecimento das placas veiculares.

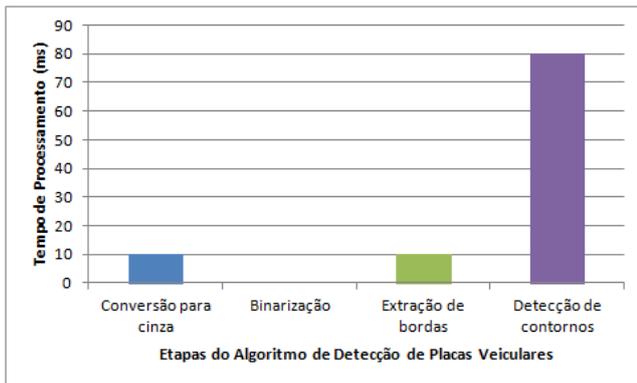
Uma taxa de acerto de 60% foi obtida pelo algoritmo para o reconhecimento dos caracteres das placas veiculares brasileiras. Enquanto essa taxa de acerto obtida não é tão alta quanto a taxa de acerto do algoritmo de detecção de placas veiculares, ela representa um valor razoavelmente bom se consideradas todas as dificuldades existentes para o reconhecimento correto de um caractere em uma placa veicular. As principais dificuldades encontradas para o reconhecimento são:

- A condição da imagem segmentada com a placa veicular: Condições de iluminação e ruído da imagem podem prejudicar o algoritmo de reconhecimento da placa veicular (Figura 10);
- A qualidade da detecção da placa veicular: Quanto maior a quantidade de *pixels* que não consistem em um caractere estiverem presentes na imagem segmentada da placa, maior a chance do algoritmo identificar algum caractere incorreto na imagem (Figura 8);
- As letras e números ambíguos: que, devido à natural similaridade da forma dos caracteres, pode confundir o algoritmo de reconhecimento de caracteres (Tabela 1);

### 5.3 Tempo de Execução

A abordagem proposta foi avaliada em termos de tempo de processamento a partir da aplicação do algoritmo sobre as imagens do *dataset* utilizado.

A média de tempo de processamento obtida para as imagens pode ser vista na Figura 16 para as duas etapas principais do algoritmo: a detecção e o reconhecimento da placa veicular. O algoritmo de detecção necessita de um tempo de execução duas vezes maior do que o algoritmo de reconhecimento, muito por conta do fato de que o algoritmo de detecção opera sobre imagens de resolução maior do que o algoritmo de reconhecimento de caracteres, que já opera sobre a imagem com a placa segmentada. Além disso, o tempo de execução total obtido pelo algoritmo proposto (150 milissegundos) é compatível com execuções interativas, sendo



**Figura 17:** Tempo de processamento obtido (em milissegundos) para as etapas do algoritmo de detecção de placas veiculares.

bem mais rápido do que a maioria dos algoritmos propostos na literatura, que obtém tempo de processamento na ordem dos segundos.

Em comparação com à abordagem proposta em [9], que obtém tempo de execução em tempo real (na ordem de 30 a 60 milissegundos) para o algoritmo de detecção da placa veicular, o algoritmo de detecção proposto neste documento é mais lento que o algoritmo proposto em [9], mas, assim como discutido na Seção 5.2, o algoritmo aqui proposto possui menor detecção de falsos positivos, foi validado com mais imagens, e se provou mais geral do que a abordagem proposta em [9], uma vez que possui boa taxa de precisão tanto para a detecção de placas veiculares brasileiras, quanto para a detecção de placas estrangeiras.

Na Figura 17, é mostrado o desempenho médio obtido por cada uma das etapas do algoritmo de detecção da placa veicular. Os algoritmos de conversão para tons de cinza e detecção de bordas precisaram de apenas 10 milissegundos para o processamento da imagem. O algoritmo de binarização precisou de menos de 0.1 milissegundos para execução, por isso praticamente não aparece no gráfico da Figura 17. A etapa mais custosa do algoritmo de detecção consiste na detecção dos contornos e seleção da localização da placa veicular (mostrados como uma única etapa na Figura 17). Percorrer o algoritmo com as bordas detectadas de forma a detectar todos os contornos retangulares presentes na imagem é computacionalmente custoso, ainda mais quando esta operação é feita sobre imagens de alta resolução, assim como podem ser caracterizadas as imagens contidas no *dataset* por conta da alta quantidade de *pixels* presentes na imagem.

Uma possível otimização para o algoritmo de detecção da placa veicular seria a adaptação do algoritmo para execução paralela nas unidades de processamento gráfico (GPUs), com exceção ao método de detecção de contornos, que é nativamente sequencial. Essa abordagem foi estudada, contudo o uso de uma abordagem híbrida com o uso do processamento paralelo não geraria melhoria significativa no tempo de processamento da aplicação.

Para o algoritmo de reconhecimento de caracteres, a etapa

mais custosa, que consome quase todo o tempo de processamento do algoritmo, é o reconhecimento dos caracteres feito internamente pela biblioteca *Tesseract*. Nesse sentido, pouca otimização pode ser feita para reduzir o tempo de processamento desta etapa, uma vez que o método de reconhecimento dos caracteres é fechado na própria biblioteca. A etapa seguinte de processamento dos caracteres reconhecidos para desambiguação e retorno dos sete caracteres da placa veicular brasileira demanda menos de 2 milissegundos para execução.

Diferentemente das etapas do algoritmo de detecção de placas, o uso de uma abordagem paralela para otimização do algoritmo de reconhecimento de caracteres não é trivial, por isso não foi integrado à abordagem proposta.

## 6. CONSIDERAÇÕES FINAIS

### 6.1 Conclusão

Neste trabalho foi apresentada uma nova técnica para detecção e reconhecimento automático de placas veiculares em tempo real. Para a detecção de placas veiculares, uma abordagem baseada em operações de binarização, extração de bordas e detecção de contornos foi utilizada para garantir uma alta taxa de precisão na localização da placa veicular, enquanto mantendo tempo de processamento interativo para o algoritmo. Para o reconhecimento interativo das placas veiculares, a placa localizada na etapa anterior foi segmentada e um algoritmo específico de reconhecimento de caracteres foi aplicado de forma a localizar os caracteres alfanuméricos da placa. Pós-processamento foi realizado de forma a adequar os caracteres detectados às restrições da placa veicular brasileira.

Os resultados foram obtidos após a avaliação das técnicas em um *dataset* com centenas de imagens obtidas a partir de placas veiculares produzidas não somente no Brasil, como em outros países. Tempo de processamento foi medido em um computador de configuração convencional. Foi mostrado que as técnicas propostas neste documento obtém boa precisão tanto para a detecção quanto para o reconhecimento da placa veicular, enquanto mantendo baixo tempo de processamento para as técnicas.

### 6.2 Trabalhos Futuros

Para melhorar o desempenho da abordagem proposta, uma adaptação do algoritmo para execução paralela em GPUs poderia ser estudada. Para a detecção das placas veiculares, uma abordagem híbrida entre a abordagem aqui proposta e o algoritmo proposto em [9] poderia ser implementada e avaliada, contudo, deveria-se garantir que essa abordagem híbrida reduzisse o tempo de processamento enquanto mantivesse a precisão obtida pela abordagem proposta. Já para o reconhecimento de placas veiculares, enquanto o uso de GPUs não é popular nesse contexto, uma abordagem que fizesse o processamento paralelo do reconhecimento da placa por cada caractere pode ser uma alternativa inicial para otimização.

Outro ponto que pode ser feito no futuro é o uso e avaliação da abordagem proposta em cenários reais, com milhares de placas veiculares capturados em inúmeras condições de iluminação. É sabido que o uso de um *dataset* com 400 imagens tem suas limitações, uma vez que centenas de imagens

podem não capturar toda a diversidade existente na captura das placas veiculares, bem como toda a variedade de placas veiculares existentes em diferentes países.

O algoritmo de reconhecimento de caracteres foi testado apenas para as placas veiculares brasileiras, devido à diferença da fonte dos caracteres e do conteúdo dentro da placa interferir diretamente na precisão do algoritmo de reconhecimento de caracteres. Nesse sentido, o estudo e desenvolvimento de uma abordagem mais geral para reconhecimento de caracteres em placas veiculares é necessária para que a técnica atinja uma precisão que permita a sua aplicação em um cenário real.

Abordagens para a detecção e reconhecimento das placas veiculares em cenários desafiadores em que há oclusão (visualização parcial) da placa veicular, em que a placa veicular está danificada, entre outros cenários, também é desejável para melhorar ainda mais a precisão obtida no reconhecimento automático da placa veicular.

## 7. REFERÊNCIAS

- [1] Medialab Dataset. <http://www.medialab.ntua.gr/research/LPRdatabase.html/>, 2002.
- [2] ABDULLAH, S. N. H. S., KHALID, M., YUSOF, R., AND OMAR, K. License plate recognition using multi-cluster and multilayer neural networks. In *Proceedings of the 2nd International Conference on Information Communication Technologies.* (2006), vol. 1, pp. 1818–1823.
- [3] AGHDASI, F., AND NDUNGO, H. Automatic licence plate recognition system. In *Proceedings of the 7th Conference in Africa (AFRICON)* (Sept 2004), vol. 1, pp. 45–50 Vol.1.
- [4] AL-GHAILI, A. M., MASHOHOR, S., ISMAIL, A., AND RAMLI, A. R. A new vertical edge detection algorithm and its application. In *Proceedings of the International Conference on Computer Engineering Systems (ICCES)* (Nov 2008), pp. 204–209.
- [5] BRADSKI, G., AND KAEHLER, A. *Learning OpenCV: Computer Vision in C++ with the OpenCV Library*, 2nd ed. O’Reilly Media, Inc., 2013.
- [6] CANNY, J. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence PAMI-8*, 6 (Nov 1986), 679–698.
- [7] DU, S., IBRAHIM, M., SHEHATA, M., AND BADAWY, W. Automatic license plate recognition (alpr): A state-of-the-art review. *IEEE Transactions on Circuits and Systems for Video Technology* 23, 2 (Feb 2013), 311–325.
- [8] DUAN, T. D., DUC, D. A., AND DU, T. L. H. Combining hough transform and contour algorithm for detecting vehicles’ license-plates. In *Proceedings of the International Symposium on Intelligent Multimedia, Video and Speech Processing.* (Oct 2004), pp. 747–750.
- [9] FILHO, J. F. S. P. Localização de placas de licenciamento veicular em tempo real utilizando opencv com cuda. Trabalho de conclusao de curso, Instituto Federal da Bahia, 2014.
- [10] GASTAL, E. S. L., AND OLIVEIRA, M. M. Domain transform for edge-aware image and video processing. *ACM TOG* 30, 4 (2011), 69:1–69:12. Proceedings of SIGGRAPH 2011.
- [11] GONCALVES, G., SILVA, S., MENOTTI, D., AND SCHWARTZ, W. A benchmark for license plate character segmentation. In *arXiv preprint* (2016).
- [12] GONZALEZ, R., AND WOODS, R. *Processamento de Imagens Digitais*. Edgard Blucher, 2000.
- [13] GUINGO, B., RODRIGUES, R., AND THOME, A. Reconhecimento automatico de placas de veiculos automotores atraves de redes neurais artificiais. In *Congresso Brasileiro de Computação* (2002).
- [14] HO, W. T., LIM, H. W., AND TAY, Y. H. Two-stage license plate detection using gentle adaboost and sift-svm. In *Proceedings of the First Asian Conference on Intelligent Information and Database Systems (ACIIDS)* (April 2009), pp. 109–114.
- [15] HU, P., ZHAO, Y., YANG, Z., AND WANG, J. Recognition of gray character using gabor filters. In *Proceedings of the 5th International Conference on Information Fusion.* (July 2002), vol. 1, pp. 419–424 vol.1.

- [16] KAMAT, V., AND GANESAN, S. An efficient implementation of the hough transform for detecting vehicle license plates using dsp's. In *Proceedings of the Real-Time Technology and Applications Symposium* (May 1995), pp. 58–59.
- [17] KANAYAMA, K., FUJIKAWA, Y., FUJIMOTO, K., AND HORINO, M. Development of vehicle-license number recognition system using real-time image processing and its application to travel-time measurement. In *Proceedings of the IEEE Vehicular Technology Conference* (May 1991), pp. 798–804.
- [18] KIM, K. K., KIM, K. I., KIM, J. B., AND KIM, H. J. Learning-based approach for license plate recognition. In *Proceedings of the IEEE Signal Processing Society Workshop*. (2000), vol. 2, pp. 614–623 vol.2.
- [19] LEE, E. R., KIM, P. K., AND KIM, H. J. Automatic recognition of a car license plate using color image processing. In *Proceedings of the International Conference on Image Processing (ICIP)* (Nov 1994), vol. 2, pp. 301–305 vol.2.
- [20] LEE, H.-J., CHEN, S.-Y., AND WANG, S.-Z. Extraction and recognition of license plates of motorcycles and vehicles on highways. In *Proceedings of the 17th International Conference on Pattern Recognition (ICPR)* (Aug 2004), vol. 4, pp. 356–359 Vol.4.
- [21] LI, T., AND YU, S. Parallel license plate recognition scheme based on image splicing for intelligent transportation system. In *Proceedings of the Fourth International Conference on Computer Science and Network Technology (ICCSNT)* (Dec 2015), vol. 01, pp. 1482–1485.
- [22] LIU, G., MA, Z., DU, Z., AND WEN, C. The calculation method of road travel time based on license plate recognition technology. In *Proceedings of the International Conference on Advances in Information Technology and Education* (Berlin, Heidelberg, 2011), H. Tan and M. Zhou, Eds., Springer Berlin Heidelberg, pp. 385–389.
- [23] LLORENS, D., MARZAL, A., PALAZÓN, V., AND VILAR, J. M. *Car License Plates Extraction and Recognition Based on Connected Components Analysis and HMM Decoding*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2005, pp. 571–578.
- [24] MATAS, J., AND ZIMMERMANN, K. Unconstrained licence plate and text localization and recognition. In *Proceedings of the IEEE Intelligent Transportation Systems, 2005*. (Sept 2005), pp. 225–230.
- [25] OTSU, N. A threshold selection method from gray-level histograms. *Proceedings of the IEEE Transactions on Systems, Man, and Cybernetics* 9, 1 (Jan 1979), 62–66.
- [26] RAHMAN, C. A., BADAWY, W., AND RADMANESH, A. A real time vehicle's license plate recognition system. In *Proceedings of the IEEE Conference on Advanced Video and Signal Based Surveillance*. (July 2003), pp. 163–166.
- [27] SAKURAI, C., FONTANA, C., YOSHIOKA, L., MARTE, C., AND SANTOS, A. License plate recognition as a tool for fiscal inspection. In *Proceedings of the 21st World Congress on Intelligent Transport Systems* (2014), pp. 360–371.
- [28] SANYUAN, Z., MINGLI, Z., AND XIUZI, Y. Car plate character extraction under complicated environment. In *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, (2004), vol. 5, pp. 4722–4726 vol.5.
- [29] SARFRAZ, M. S., SHAHZAD, A., ELAHI, M. A., FRAZ, M., ZAFAR, I., AND EDIRISINGHE, E. A. Real-time automatic license plate recognition for cctv forensic applications. *Journal of Real-Time Image Processing* 8, 3 (2013), 285–295.
- [30] SHI, X., ZHAO, W., AND SHEN, Y. *Automatic License Plate Recognition System Based on Color Image Processing*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2005, pp. 1159–1168.
- [31] SIRITHINAPHONG, T., AND CHAMNONGTHAI, K. The recognition of car license plate for automatic parking system. In *Proceedings of the Fifth International Symposium on Signal Processing and Its Applications (ISSPA)* (1999), vol. 1, pp. 455–457 vol.1.
- [32] SMITH, R. An overview of the tesseract ocr engine. In *Proceedings of the 9th International Conference on Document Analysis and Recognition (ICDAR)* (Washington, DC, USA, 2007), IEEE Computer Society, pp. 629–633.
- [33] SOBEL, I. An isotropic 3x3 image gradient operator. 376–379.
- [34] SUZUKI, S., AND BE, K. Topological structural analysis of digitized binary images by border following. *Computer Vision, Graphics, and Image Processing* 30, 1 (1985), 32 – 46.
- [35] YOIMORI, S., MITSUKURA, Y., FUKUMI, M., AKAMATSU, N., AND PEDRYCZ, N. License plate detection system by using threshold function and improved template matching method. In *Proceedings of the IEEE Annual Meeting of the Fuzzy Information*. (June 2004), vol. 1, pp. 357–362 Vol.1.
- [36] ZHENG, D., ZHAO, Y., AND WANG, J. An efficient method of license plate location. *Pattern Recognition Letters* 26, 15 (2005), 2431 – 2438.