CEEL - ISSN 2596-2221
Universidade Federal de Uberlândia
07 a 11 de dezembro de 2020

DETECÇÃO DE PADRÕES EM IMAGENS ATRAVÉS DE HISTOGRAMAS DE GRADIENTES ORIENTADOS E CLASSIFICADORES LINEARES DO TIPO SVM

Maycow dos Santos*[1] e Milena Bueno Pereira Carneiro[1]

[1]FEELT – Universidade Federal de Uberlândia

Resumo – Esse artigo apresenta algumas aplicações práticas que utilizam o método de histograma de gradientes orientados em conjunto com máquinas de vetores de suporte com o intuito de exemplificar e difundir o poder que a união desses métodos possui no campo de visão computacional e identificação de padrões em imagens. As primeiras seções do artigo possuem o objetivo de apresentar de maneira suscinta o que é o histograma de gradientes orientados e o que é um classificador SVM (Máquina de vetores de suporte) para então, mostrar aplicações de detecção facial, detecção de pedestres e detecção de veículos.

***Palavras-Chave* – Aprendizado de máquina, Histograma de gradientes orientados, Máquina de vetores de suporte, Processamento de imagens, Reconhecimento de padrões, Visão computacional**

IMAGE PATTERN DETECTION WITH HISTOGRAM OF ORIENTED GRADIENTS AND SVM LINEAR CLASSIFIER

***Abstract* - The main objective of this article revolves around presenting some practical applications of the union between histogram of oriented gradients and support vector machines in order to exemplify and disseminate the power that the union of these methods has in the field of computer vision and pattern identification in images. The first sections of the article aim to briefly present what is the histogram of oriented gradients and what is an SVM classifier to then show applications of facial detection, pedestrian detection and vehicle detection.**

***Keywords* – Computer vision, Histogram of oriented gradients, Image processing, Machine learning, Pattern recognition, Support vectors machine**

I. INTRODUÇÃO

Seres humanos são capazes de detectar objetos baseando-se em características como tamanho, formato, cor, etc. Imagine que um ser humano precise detectar um pedestre na rua. Essa detecção se daria após observação da presença de um formato cabeça na parte de cima, braços nas laterais e pernas embaixo. Agora imagine que um ser humano precise diferenciar um carro de um caminhão. Tal classificação se daria pela diferença de tamanho das rodas, da lataria, do formato padrão do veículo e etc.

Esses parâmetros capturados pelos olhos e processados pelo cérebro humano nem sempre são os ideais para a visão computacional. Uma imagem colorida com um amontoado de pixels dispostos de uma maneira qualquer com processamento pixel a pixel não fornece informações suficientes para que um computador reconheça e classifique um determinado objeto. Isso faz com que seja necessário o desenvolvimento de técnicas específicas para transformar uma imagem em um conjunto de dados que sejam compatíveis com o modelo de processamento de imagens para visão computacional.

Em 2005 foi proposto por Dalal e Triggs, em seu artigo "*Histograms of Oriented Gradients for Human Detection*", um método muito interessante para a detecção de pedestres baseado em transformar uma imagem em um conjunto de dados que trazem informações importantes para a detecção de objetos através das variações de contraste na imagem[1]. Embora esse método tenha sido utilizado por Dalal e Triggs para a detecção de pedestres, é possível estender o uso dessa metodologia para aplicações diversas como, por exemplo, a detecção facial ou reconhecimento de objetos em geral.

A ideia básica por trás da detecção de um objeto através de histogramas de gradientes orientados é utilizar um grande banco de imagens com duas classes: Imagens que contenham o objeto em questão e imagens que não contenham o objeto em questão. Então, é possível extrair os histogramas de gradientes orientados de todas as imagens de treinamento e utilizar todos esses dados para treinar um classificador, de modo que, toda vez que uma nova imagem for submetida para detecção, o modelo treinado realize a extração das características HOG (Histogramas de gradientes orientados) da nova imagem e submeta essas características ao classificador treinado com milhares de imagens. Através desse processo é possível treinar modelos de detecção que podem ser aplicados em uma variedade de áreas: carros autônomos,

*maycowdossantos@hotmail.com

reconhecimento facial, imagens médicas, geoprocessamento, setor agropecuário, etc.

As seções 2 e 3 desse artigo trazem, de maneira breve, uma explicação a respeito do método dos histogramas de gradientes orientados e dos classificadores Máquina de Vetores Suporte (*Support Vectors Machine* - SVM). A seção 4 exemplifica a aplicação prática desses métodos através de três aplicações: detecção facial, detecção de pedestres e detecção de veículos.

II. HISTOGRAMAS DE GRADIENTES ORIENTADOS

O Histograma de Gradientes Orientados (HGO) é um descritor de recurso proposto por Dalal e Triggs em 2005 que tinha como objetivo realizar a detecção de pedestres baseando-se na variação de iluminação de uma imagem [1]. De maneira simples: Dado um pixel em uma imagem em escala de cinza, observa-se os pixels na vizinhança imediata e os compara com o pixel analisado, de modo a determinar o quão mais claro ou escuro eles são entre si. Dessa maneira, utiliza-se um vetor para indicar a direção onde a imagem fica mais escura. Após realizar esse processo para todos os pixels da imagem, é possível realizar o cálculo dos histogramas dos gradientes e utilizá-los como dados de treinamento de um classificador linear SVM (Máquina de vetores de suporte).
Antes da compreensão das etapas do método proposto por Dalal e Triggs, é importante explanar os conceitos de gradiente e histograma, para então aglutinar ambos e formar a ideia básica do histograma de gradientes orientados.

A. Gradientes orientados

O gradiente de uma imagem é um vetor que indica a direção e a variação dos níveis de cinza de uma imagem, esse vetor aponta para a direção da maior variação de uma função f nas coordenadas (x,y) [2]. O vetor gradiente de uma imagem em uma determinada posição (x,y) pode ser calculado pela Equação 1 abaixo.

$$\nabla \mathbf{f}(x,y) = \frac{\partial f(x,y)}{\partial x}\,\mathbf{i} + \frac{\partial f(x,y)}{\partial y}\,\mathbf{j} \qquad (1)$$

Para simplificar a notação, a primeira parcela da Equação 1 será chamada de Gx e a segunda parcela da Equação 1 será chamada de Gy, onde Gx e Gy representam o gradiente da imagem nas direções x e y respectivamente. A Equação 2 mostra o cálculo da magnitude do operador gradiente e a Equação 3 mostra o cálculo da orientação do vetor gradiente.

$$Mag(\nabla f) = \sqrt{G_x^2 + G_y^2} \qquad (2)$$

$$\theta(\nabla f) = \arctan\left(\frac{Gy}{Gx}\right) \qquad (3)$$

B. Processo de obtenção das características do HOG

A primeira etapa do método é uma espécie de pré-processamento das imagens que consiste basicamente em aplicar uma correção gama, realizar corte e redimensionamento das imagens. A correção gama aplicada, segundo o autor, não mostrou uma melhora significativa na performance do método por conta das etapas posteriores de normalização que atingem resultados semelhantes.

A segunda etapa do método consiste em computar os gradientes das imagens pré-processadas. Essa etapa consiste em filtrar a imagem com alguma máscara (máscaras de Sobel 3x3, máscaras 2x2 ou até máscara unidimensional do tipo linha [-1, 0, 1] na direção x e do tipo coluna com esses mesmos valores na direção y). Dessa maneira é possível obter o vetor gradiente em um dado pixel e é possível calcular a magnitude e a fase desse vetor gradiente através das Equações 2 e 3 respectivamente.

A terceira etapa do método consiste em dividir a imagem em células de um determinado tamanho (8x8, 4x4 e etc.) e um histograma de gradientes é calculado para cada uma dessas células da imagem, de modo que se obtém um vetor com nove posições que representa o histograma obtido para cada célula. Cada posição desse vetor diz respeito a uma faixa de ângulos e o valor do vetor nessa determinada posição diz respeito a quantidade de gradientes que estão orientados nessa determinada faixa de ângulos.

A quarta etapa consiste em normalizar os histogramas obtidos de modo que eles se tornem menos sensíveis a variações de iluminação na imagem. Um bloco maior que as células pode ser utilizado para essa normalização, tornando o processo mais robusto e menos suscetível a ruídos na imagem. A saída dessa etapa fornece um vetor que representa o histograma de gradientes orientados da imagem. Assim, a quinta etapa consiste em utilizar esses histogramas para treinar um classificador SVM.

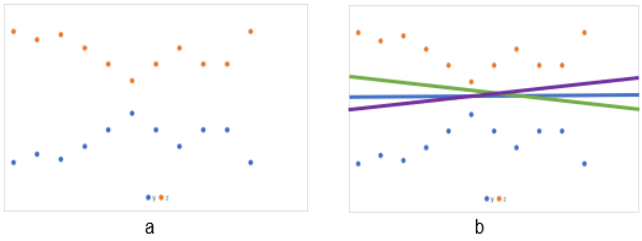
III. MÁQUINA DE VETORES DE SUPORTE

O classificador Máquina de Vetores Suporte (SVM) é um algoritmo de classificação e regressão cujo objetivo na área de classificação é separar classes denominadas linearmente separáveis através de um processo de aprendizado estatístico [3]. Classes linearmente separáveis são classes que podem ser separadas através de uma reta e, mesmo quando é impossível separar classes através de retas, é possível utilizar o modelo SVM através de um truque de kernel, que consiste basicamente em modificar o algoritmo para que sejam utilizadas funções de separação curvilíneas (polinomial, gaussiana e etc).

Na Figura 1.a abaixo é possível observar um exemplo de duas classes de dados quaisquer que podem ser separados por uma linha. A Figura 1.b mostra as várias possíveis retas que poderiam ser utilizadas para separar essas duas classes, no entanto, o modelo SVM calcula a linha de separação entre as classes de maneira que a distância entre a reta e os pontos extremos das classes seja maximizada, de modo que seja possível obter uma separação mais generalista e que possua ótima aplicabilidade à novos dados de entrada para qualquer classe.

Denomina-se vetor de suporte o vetor que tenha origem na linha de separação entre as classes e fim no ponto mais próximo da linha de separação para cada uma das classes e denomina-se margem toda a área entre o vetor de suporte e a linha de separação.

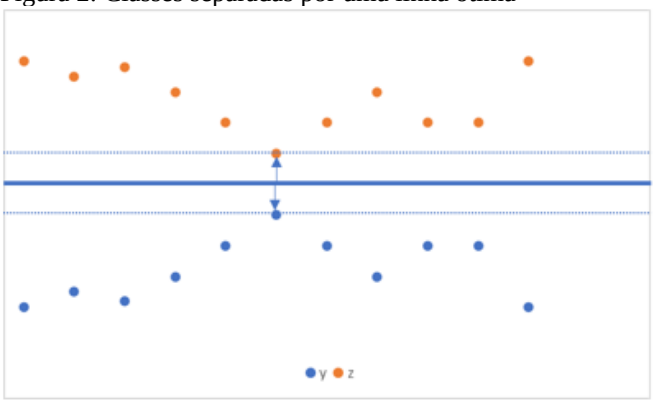
Figura 1: Exemplo de classes linearmente separáveis



Fonte: O Autor

A Figura 2 mostra duas classes separadas por uma linha que otimize a margem com seus respectivos vetores de suporte representados.

Figura 2: Classes separadas por uma linha ótima



Fonte: O Autor

Os classificadores SVM lineares podem possuir margens rígidas ou margens suaves. Os classificadores de margens rígidas realizam a otimização da separação entre as classes através da imposição de restrições que asseguram que não haja dados de treinamento entre as margens de separação das classes [3]. Essas restrições nem sempre podem ser atendidas devido à natureza dos dados e, em casos onde duas ou mais classes não são completamente separadas por uma entidade linear, se faz necessário o uso de um classificador SVM de margens suaves, que mostra maior tolerância a ruídos e trabalha de maneira amena em relação às restrições impostas pelos classificadores de margens rígidas.

IV. APLICAÇÕES PRÁTICAS

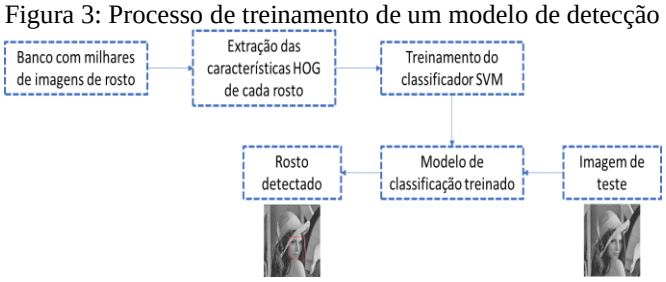
As seções anteriores desse artigo apresentaram uma visão geral a respeito dos métodos e algoritmos fundamentais para a realização das aplicações práticas que serão mostradas nessa seção.

A. Detecção facial

A detecção facial consiste em identificar rostos em uma imagem e devolver como parâmetros de saída as posições desses rostos, podendo ser utilizada como uma das etapas de reconhecimento facial, para aplicação de filtros automáticos sobre o rosto, para alinhamento facial, para melhorar a aplicação automática de foco em câmeras digitais, etc. Como é possível perceber, a detecção facial é uma peça chave para várias outras aplicações importantes e úteis.

O processo mais fácil para treinar um modelo de detecção facial se dá através de um processo de aprendizado supervisionado que consiste em utilizar um banco de dados com várias imagens de rostos. A partir desse banco de imagens é realizada a extração dos histogramas de gradientes orientados de todos os rostos. Esses histogramas são submetidos a um classificador SVM para que esse classificador aprenda a identificar qual o padrão de um rosto baseado em suas características de variação de iluminação. Com um modelo treinado em mãos, basta submeter uma nova imagem ao classificador previamente treinado. A imagem submetida para classificação pode ser repartida em blocos ou janelas de tamanhos variáveis, de modo que cada bloco da imagem seja classificado pelo modelo treinado. Através desse processo os vários blocos da imagem são classificados como "face" ou "não-face", então, o resultado é uma sobreposição do resultado de classificação de cada bloco da imagem, resultando na determinação da posição de um rosto na imagem de teste. A Figura 3 resume o processo de treinamento e classificação de um modelo de detecção facial. A Figura 4.a exemplifica a classificação obtida sobre cada bloco da imagem de teste e a Figura 4.b exibe o resultado do processo de classificação através da sobreposição dos resultados obtidos na Figura 4.a.

Figura 3: Processo de treinamento de um modelo de detecção



Fonte: O Autor

Figura 4: Exemplo de classificação de blocos e resultado

Fonte: O Autor

Um modelo útil para avaliação dos resultados de assertividade das detecções faciais consiste em utilizar um conjunto de dados de consulta no qual já se sabe quais imagem possuem uma ou mais faces e quais as posições dessas faces em cada

imagem [4], submetendo essas imagens a um modelo treinado é possível comparar as previsões do modelo de detecção facial com os resultados reais, obtendo então a taxa de acertos do modelo de detecção. Cerca de 20 % das imagens dos bancos de imagens utilizados serviram como imagens de teste e 80 % serviram como imagens de treinamento. Com base em 5 conjuntos diferentes de imagens [5, 6, 7, 8 e 9] é possível verificar que a taxa de acertos nas predições através do método HOG foram de 92 % para conjunto 1, 94 % para o conjunto 2, 87 % para o conjunto 3, 90 % para o conjunto 4 e 89 % para o conjunto 5, com média 90.4 % de acertos, a Tabela 1 sintetiza os resultados obtidos para as imagens de teste de cada *dataset* utilizado.

Tabela 1: Resumo das taxas de acerto obtidas

Dataset	**Taxa de acertos**
Face 94 [5]	92 %
Face 95 [6]	94 %
Face 96 [7]	87 %
Grimace [8]	90 %
PICS [9]	89 %
Média	90.4 %

O primeiro conjunto possui 3078 imagens de 15 pessoas diferentes em um fundo verde, sem variações de escala, sem variações na posição do rosto na imagem e sem variações de iluminação [5]. O segundo conjunto possui 1440 imagens de 72 pessoas diferentes, o plano de fundo é uma cortina vermelha, as imagens possuem variações na escala do rosto, com variações de iluminação e sem muita variação da posição dos rostos nas imagens [6]. O terceiro conjunto de imagens possui 3016 imagens de 152 pessoas diferentes, os fundos das imagens são coloridos e variados, existe variação de escala nas faces, possui poucas variações na inclinação dos rostos e possui mudanças significativas de iluminação [7]. O quarto conjunto possui 360 imagens de 18 indivíduos diferentes, o fundo das imagens é simples, sem muitas variações de escala nos rostos, altas variações na inclinação e posição das faces nas imagens e possui pouca variação de iluminação [8]. O quinto conjunto possui 599 imagens de 23 indivíduos diferentes, com plano de fundo constante, com pouca variação de iluminação e alta variação na inclinação e posição das faces na imagem.

A Figura 5 exibe uma amostra de imagem de cada um dos conjuntos de imagens citados acima, a Figura 5.a mostra o primeiro *dataset* [5], a Figura 5.b mostra o segundo *dataset* [6], a Figura 5.c mostra o terceiro *dataset* [7] , a Figura 5.d mostra o quarto *dataset* [8] e a Figura 5.e mostra o quinto *dataset* [9]. A Figura 6 exibe um exemplo de rostos detectados através do processo descrito nessa seção.

Figura 5: Amostra de imagens de cada um dos conjuntos

Fonte: Face [5,6 e 7], Grimace [8] e PICS [9]

Figura 6: Rostos detectados através do processo HOG

Fonte: Adaptado de br.freepik.com

B. Detecção de pedestres

A detecção de pedestres consiste em identificar, em uma imagem estática ou vídeo, uma região da imagem que possua atribuições e formato humano. Tal detecção é fundamental para sistemas veiculares autônomos, pois esses veículos precisam monitorar continuamente o ambiente ao seu redor para garantir o menor nível de risco à vida.

Para o método de detecção de pedestres desenvolvido por Dalal e Triggs, o principal banco de imagens utilizados foi o *dataset* chamado INRIA que possui 1805 imagens de pessoas em várias poses, orientações, diversos planos de fundo, em meio a carros e multidões [10]. De algumas partes das imagens positivas (imagens que contém pelo menos um pedestre) foram extraídas as imagens negativas (que não possuem pedestres) com o intuito de extrair as características HOG das imagens de pedestres e de não pedestres, para então treinar um classificador SVM.

A avaliação de performance do método se dá através da observação da relação entre a taxa de erros dada pela Equação 4 e a quantidade de falsos positivos por janela observados na etapa de teste do método. Após a realização de testes em 100 imagens foi possível obter uma taxa de acertos de 89 %.

$$\frac{Falsos\ negativos}{Verdadeiros\ positivos + falsos\ negativos} \quad (4)$$

A Figura 7 e a Figura 8 exibem imagens de diversos pedestres detectados.

Figura 7: Pedestres detectados em um contexto urbano

Fonte: Adaptado de facility.org.br

Figura 8: Diversos pedestres detectados

Fonte: Adaptado de carrodegaragem.com

C. Detecção de veículos

A detecção de veículos via visão computacional consiste na aplicação de técnicas de processamento de imagem e treinamento de modelos de aprendizado de máquina para identificar veículos em imagens. A detecção veicular é uma tecnologia importante para que veículos autônomos possam identificar outros veículos trafegando na via, para que possam estacionar de maneira autônoma, para sistemas de segurança aéreo etc. Foi utilizado o banco de imagens GTTI fornecido pelo grupo de tratamento de imagens da Universidade Politécnica de Madri. Esse banco possui 5966 imagens de veículos em diversas posições, de diversas cores e de diversos tamanhos e 5068 imagens que não são de veículos e foram extraídas do céu, de vegetações, de placas de trânsito e do chão pois são cenas comuns em imagens que contêm veículos, mas não são veículos [11]. A Figura 9 exibe algumas imagens positivas do *dataset* em questão e a Figura 10 exibe algumas imagens negativas desse *dataset*.

Figura 9: Exemplos de imagens positivas de treinamento

Fonte: GTTI [11]

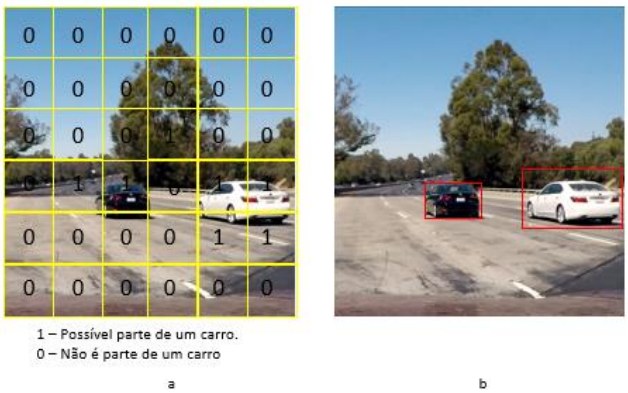
Figura 10: Exemplos de imagens negativas de treinamento

Fonte: GTTI [11]

O processo geral para o treinamento de um modelo de detecção veicular consiste em várias etapas, a primeira etapa visa extrair as características de HOG de todas as imagens positivas e das imagens negativas do banco de dados de treino. Após a extração das características HOG de cada imagem a etapa 2 consiste em treinar um classificador SVM com os dados obtidos na etapa 1, esse classificador se torna capaz de diferenciar veículos de não veículos baseando-se no padrão HOG obtido através das imagens de treinamento [12].

A terceira etapa do processo consiste em implementar uma janela deslizante que é capaz de varrer sub-regiões de uma imagem em um frame de um vídeo ou de uma imagem estática para que o modelo treinado na etapa 2 realize a classificação em cada uma das sub-regiões da imagem, após a varredura completa da imagem o resultado final se dá através da sobreposição do resultado da análise de cada sub-região. A Figura 11.a exemplifica a classificação obtida sobre cada bloco da imagem de teste e a Figura 11.b exibe o resultado do processo de classificação através da sobreposição dos resultados obtidos na Figura 11.a.

Figura 11: Exemplo de classificação via janela deslizante



Fonte: Adaptado de GTTI [11]

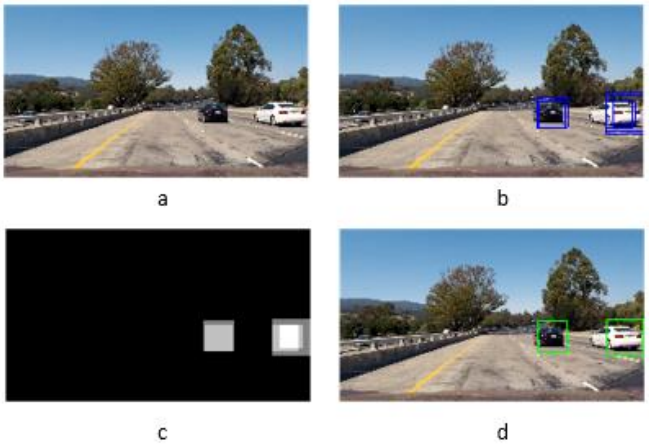
Quando esse processo é aplicado em uma imagem podem surgir falsos positivos em regiões da imagem que não representam veículos. Esses falsos positivos podem ser prejudiciais para a tomada de decisões em tempo real em veículos autônomos devido às velocidades nas quais as decisões precisam ser tomadas. Ao analisar as detecções de frames consecutivos de um vídeo que exibe veículos

percorrendo uma rodovia é possível notar que grande parte das classificações são verdadeiros positivos e que os falsos positivos são pontuais e esporádicos.

Com o intuito de aumentar a confiabilidade nos resultados e contornar os efeitos da presença esporádica de falsos negativos é interessante implementar um mapa de calor de classificação, a cada frame classificado ocorre o processo de "adicionar uma unidade de calor" sobre as janelas que foram classificadas como positivas, após a sobreposição dos resultados obtidos em alguns frames é possível diferenciar áreas mais "quentes" e áreas mais "frias". As áreas quentes do mapa de calor simbolizam áreas da imagem onde várias janelas foram classificadas como positivas, as áreas frias do mapa indicam onde locais da imagem onde as janelas foram classificadas como negativas para a presença de veículos. Quanto mais janelas classificadas como positivas em frames consecutivos mais quente será a região da imagem no domínio do mapa de calor. Adicionando um limiar de "temperatura" no mapa de calor é possível excluir os falsos negativos que ocorrem esporadicamente, aumentando então a confiabilidade da classificação sobre frames consecutivos.

A Figura 12.a exibe uma imagem de teste, a Figura 12.b exibe em azul várias as janelas deslizantes que foram classificadas como positivas, a Figura 12.c exibe o mapa de calor obtido através da sobreposição dos resultados das várias janelas deslizantes positivas e a Figura 12.d exibe a classificação final, que é a demarcação da posição dos veículos na imagem de acordo com as áreas mais quentes no mapa de calor.

Figura 12: Imagem de teste, mapa de calor e classificação final



Fonte: Adaptado de GTTI [11]

V. CONCLUSÕES

A partir de todos os exemplos de aplicações do método HOG+SVM é possível perceber que o processo geral para a detecção de um determinado objeto é relativamente padronizado: Utiliza-se um banco de imagens positivas e negativas, realiza a extração das *features* HOG do banco de imagens e utiliza tais *features* extraídas para treinar um classificador SVM, desse modo obtém-se um modelo treinado de detecção. Além desse processo básico é preciso considerar a particularidade de cada aplicação e ter sensibilidade para contornar os problemas decorrentes dessas particularidades.

É possível utilizar os métodos apresentados para realizar a detecção de inúmeros objetos, o limite para o tipo de aplicação reside na criatividade do projetista do modelo treinado. A performance do método se mostrou satisfatória para as aplicações mostradas e pode, obviamente, ser melhorada para aplicações em veículos autônomos que necessitam de processos rápidos e confiáveis.

REFERÊNCIAS

[1] DALAL, N.; TRIGGS, B.. Histograms of Oriented Gradients for Human Detection. 2005 Ieee Computer Society Conference On Computer Vision And Pattern Recognition, v. 1, n. 1, p. 1-8, jun. 2005

[2] GONZALEZ, Rafael C.; WOODS, Richard E.. Processamento Digital de Imagens. 3. ed. São Paulo: Pearson, 2009. 644 p

[3] LORENA, Ana Carolina; CARVALHO, André de. Uma Introdução às Support Vector Machines., São Paulo, v. 14, n. 2, p. 43-67, jan. 2007.

[4] AHMAD, Faizan; NAJAM, Aaima; AHMED, Zeeshan. Image-based Face Detection and Recognition: "State of the Art". Ijcsi International Journal Of Computer Science Issues, [S.L.], v. 9, n. 1, p. 1-4, nov. 2012.

[5] Face Recognition Data, University of Essex, UK, Face 94, http://cswww.essex.ac.uk/mv/all faces/faces94.html. Acesso em: 28 ago. 2020

[6] Face Recognition Data, University of Essex, UK, Face 95, http://cswww.essex.ac.uk/mv/all faces/faces95.html. Acesso em: 28 ago. 2020

[7] Face Recognition Data, University of Essex, UK, Face 96, http://cswww.essex.ac.uk/mv/all faces/faces96.html. Acesso em: 28 ago. 2020

[8] Face Recognition Data, University of Essex, UK, Grimace, http://cswww.essex.ac.uk/mv/all faces/grimace.html. Acesso em: 28 ago. 2020

[9] Psychological Image Collection at Stirling (PICS), Pain Expressions, http://pics.psych.stir.ac.uk/2D_face_sets.htm/. Acesso em: 28 ago. 2020

[10] INRIA Person Dataset, http://www.pascal.inrialpes.fr/data/human/. Acesso em: 28 ago. 2020

[11] Vehicle image database, Universidad de Madrid, GTT, gti.ssr.upm.es/data/Vehicle_database.html. Acesso em: 28 ago. 2020

[12] SEVILLA, Mithi. Vehicle Detection with HOG and Linear SVM. 2017. Disponível em: https://medium.com/@mithi/vehicles-tracking-with-hog-and-linear-svm-c9f27eaf521a. Acesso em: 28 ago. 2020