Detecção do uso de celular utilizando técnicas de aprendizado de máquina

Rafael Alceste Berri
*Center of Technological Sciences (CCT)
Santa Catarina State University (UDESC)
Joinville, SC, Brazil, CEP 89219–710
Email: rafael.berri@joinville.udesc.br

Milton Roberto Heinen
†Computer Engineering Course, Campus Bagé
Federal University of Pampa (UNIPAMPA)
Bagé, RS, Brazil, CEP 96413–170
Email: milton.heinen@unipampa.edu.br

***Resumo*—Estima-se que 80% das colisões e 65% das quase colisões envolveram motoristas que não estavam prestando a devida atenção ao trânsito por três segundos antes do evento. Este artigo desenvolve um algoritmo para extração de característica que permitam identificar o uso de celular durante o ato de dirigir um veículo. Realizaram-se experimentos em 58 imagens positivas (com celular) e 60 imagens negativas (sem celular), utilizando-se para isso Máquina de Vetor de Suporte (SVM) e redes neurais Perceptron Multi-Camadas (MLP). Mostra-se mais vantajosa, para as características providas pelo algoritmo de extração, usar a Máquina de Vetor de Suporte (SVM) com kernel Sigmoide como classificador, obtendo-se assim a taxa de acerto de 94,09% para o sistema.**

I. Introdução

A distração ao volante [1], [2], ou seja, uma ação que leve o motorista a desviar a atenção da pista por alguns segundos, representa aproximadamente metade dos casos de acidentes no trânsito. O simples ato de discar um número de telefone, por exemplo, consume cerca de 5 segundos, implicando em 140 metros percorridos por um automóvel a 100 km/h [3]. Em um estudo feito em Washington por *Virginia Tech Transportation Institute* revelou, após 43 mil horas de testes, que quase 80% das colisões e 65% das quase colisões envolveram motoristas que não estavam prestando a devida atenção ao trânsito por três segundos antes do evento [4].

Em pesquisa desenvolvida apresentada por Salvador (2011), constatou-se uma redução de 40% no número de acidentes em Abu Dhabi, nos Emirados Árabes, durante três dias, em outubro de 2011, em que aparelhos de uma operadora estiveram fora do ar, sinaliza a grande parcela de responsabilidade do celular como uma distração ao volante. No Brasil o número de autuações por uso de celular aumentou 150% em cinco anos [5]. Durante uma conversa ao telefone, motoristas estão 4 vezes mais suscetíveis a acidentes, graves o suficiente para causar ferimentos [6]. O Conselho Nacional de Segurança dos EUA indica que o uso de celular e escrita de mensagens na direção causam ao menos 28% dos acidentes de trânsito naquele país [7].

Estudar formas de detectar o uso do celular durante o ato de dirigir é o primeiro passo para tentar minimizar esses números. O objetivo central desse artigo é apresentar um algoritmo para extrações de características que permitam identificar o uso do celular pelo motorista, possibilitando assim, avisá-lo e alertá-lo para retornar sua atenção exclusivamente ao carro. Busca-se ainda, testar classificadores e escolher a técnica que maximize a acurácia do sistema. A apresentação deste artigo se dá da seguinte forma: na Seção II apresentam-se trabalhos relacionados, ferramentas de suporte a classificação são apresentadas na Seção III, a Seção IV contém o algoritmo desenvolvido para extração das características, na Seção V apresentam-se experimentos efetuados em um banco de imagens e por fim conclusões são estabelecidas na Seção VI.

II. Trabalhos Relacionados

Dentre os artigos existentes na literatura o trabalho feito por Veeraraghavan et al. (2007) está mais próximo do enfoque deste artigo. O objetivo dos autores é alcançar a detecção e classificação de atividades do condutor de um automóvel utilizando visão computacional, consegue-se por meio de um algoritmo que detecta movimento relativo e da segmentação da pele do motorista [8]. Portanto, seu funcionamento depende de se obter um conjunto de *frames* (filmagem) do motorista. Possui também, a necessidade de colocar uma câmera lateralmente (em relação ao motorista) dentro do carro.

A abordagem de Yang et al. (2011) utiliza-se de sinais sonoros personalizados de alta frequência emitidos através do equipamento de som presente no carro, rede *Bluetooth*, e de um *software* em execução no celular para a captura e processamento dos sinais sonoros emitidos. O objetivo dos avisos sonoros é estimar a posição em que se encontra o celular, em outras palavras, se é o motorista que está utilizando o mesmo ou algum outro passageiro do carro. A proposta obteve precisão de classificação de mais de 90% [9]. No entanto, por se tratar de um sistema que dependa de *software* operando no celular a que se deseja identificar ligações, é preciso confiar na boa vontade do motorista em permitir que o software permaneça em funcionamento. A grande vantagem da técnica é seu funciona mesmo utilizando o celular com fones de ouvido (*Hands-free*).

A proposta apresentada por Watkins et al. (2011) é de identificar autonomamente comportamentos distraídos do motorista associados a mensagens de texto em celular. A abordagem utiliza um telefone celular programado para registrar toda e qualquer digitação feita, através desses registros consegue-se constatar as distrações [10].

Outro trabalho feito por Enriquez et al. (2009) estuda formas de detectar a pele. O algoritmo de segmentação apresentado pelos autores analisa dois espaço de cores, YCrCb e o LUX, utiliza-se das coordenada CR e U especificamente

[11]. A grande vantagem do algoritmo é a possibilidade de utilização em uma câmera simples (webcam) para aquisição como aquisição da imagem.

III. Definições Preliminares

Nesta seção, ferramentas de suporte a classificação de classes são apresentadas.

A. Support Vector Machines (SVM)

A SVM (Máquina de Vetor de Suporte) foi introduzida por Vapnik em 1995 e é uma ferramenta de classificação binária [12]. Considerando um conjunto de dados $\{(x_1, y_1), \cdots, (x_n, y_n)\}$ com dados de entrada $x_i \in R^d$ (sendo d o espaço de dimensional) e saída y com rotulagem $y \in \{-1, +1\}$. A ideia central da técnica é gerar um hiperplano ótimo para separar duas classes de objetos. O hiperplano é escolhido para maximizar separação entre das duas classes e gera-se com base nos vetores de suporte, daí o nome máquinas de vetor de suporte [13]. A fase de treinamento consiste na escolha dos vetores de suporte mediante aos dados de treinamento previamente rotulados. Na Figura 1 mostra-se uma visão geral do funcionamento do SVM.

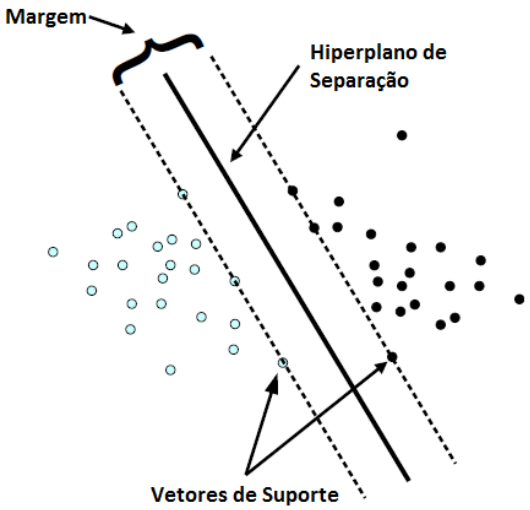

Figura 1. Esboço do SVM [14]

Com a SVM é possível utilizar algumas funções de *kernels* para tratar dados não lineares. A função do kernel transforma os dados originais em um espaço de características de alta dimensionalidade, onde as relações não lineares podem estar presentes sob a forma linear [15]. Dentre os *kernels* existentes pode-se citar: linear (Equação 1), Polinomial (Equação 2), *Radial basis function* (Equação 3) e Sigmoide (Equação 4). A escolha de uma função de adequada é um importante passo para atingir a alta acurácia do sistema de classificação.

$$K(x_i, x_j) = x_i \cdot x_j \tag{1}$$

$$K(x_i, x_j) = (\gamma(x_i \cdot x_j) + coef0)^{degree}, \gamma > 0 \tag{2}$$

$$K(x_i, x_j) = e^{-\gamma \|(x_i + x_j)\|^2}, \gamma > 0 \tag{3}$$

$$K(x_i, x_j) = tanh(\gamma(x_i \cdot x_j) + coef0) \tag{4}$$

B. Multilayer Perceptron (MLP)

Redes Neurais são sistemas computacionais cuja estrutura é inspirada por estudos do cérebro humano. Em 1958, Rosenblatt inventou o perceptron que é uma rede neural alimentação simples [13]. Com um perceptron pode-se resolver apenas problemas linearmente separáveis (classes linearmente separadas). Perceptrons com mais de uma camada de ligações variavelmente ponderados são referidos como *perceptrons* multicamada (MLP) [16]. A rede MLP pode resolver problemas não lineares e é uma ferramenta de classificação multi-classes (número de classes pode ser superior a dois).

Existem três tipos de camadas presentes em uma rede MLP [17]. A primeira camada é a chamada de entrada. Esta é responsável por receber os dados e passar para a rede, não realizando nenhum tipo de processamento. O número de neurônios na camada de entrada é igual a quantidade de variáveis de entrada. A última camada é denominada camada de saída, pois é a camada responsável por retornar a resposta calculada pela rede. A quantidade de neurônios da camada de saída depende da quantidade de saídas desejadas da rede. Entre essas camadas pode haver uma ou mais camadas intermediárias (ocultas), o número depende da complexidade do problema. Na Figura 2 mostra-se um exemplo de uma rede MLP com 4 neurônios na camada de entrada, 3 neurônios na camada intermediária e 2 neurônios na camada de saída.

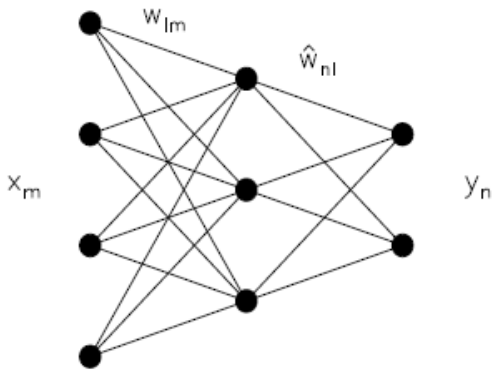

Figura 2. Um diagrama esquemático de uma rede de várias camadas e com uma camada oculta [18]

Os neurônios das camadas intermediárias e de saída realizam o processamento. Individualmente, um neurônio tem as ligações de entrada (que recebem da camada anterior) e ligações de saída (passa a resposta para a camada seguinte). Cada valor de entrada do neurônio é multiplicado pelo seu respectivo peso, os resultados são somados e adicionados ao *bias* (peso do neurônio) [16]. A soma resultante é transformada utilizando uma função f de ativação. Nas Equações 5, 6 e 7 mostram-se, respectivamente, as funções de ativação: Identidade, Sigmoide Simétrica e Gaussiana.

$$f(x) = x \tag{5}$$

$$f(x) = \frac{\beta \times (1 - e^{-\alpha \times x})}{(1 + e^{-\alpha \times x})} \tag{6}$$

$$f(x) = \beta \times e^{-\alpha x \times x} \tag{7}$$

Para que uma rede MLP consiga classificar, precisa-se passar por um treinamento. Um dos treinamentos possíveis é o denominado *backpropagation* [19]. Este processo visa descobrir os pesos (W na Figura 2) que melhoram a acurácia da rede e se dá pela diminuição gradual do erro da rede. Os erros são estimados utilizando-se os valores de treinamentos já previamente rotulados.

IV. Extração de Características

O ato de dirigir faz o motorista olhar instintiva e continuamente para os lados, porém, com a utilização do telefone celular, a tendência é de fixar o olhar em um ponto à frente, afetando a dirigibilidade e a atenção do condutor no trânsito, por limitar o campo de visão [20]. Partindo-se desse princípio optou-se por desenvolver um algoritmo que consiga perceber a utilização do celular pelo motorista apenas à frente de uma câmera acoplada no painel de um carro. As subseções seguintes explicam seu funcionamento.

A. Algoritmo

De maneira geral o algoritmo está dividido nas seguintes partes:

- **Aquisição:** Abertura do arquivo da imagem.
- **Pré-processamento:** Localização do motorista, corte da região de interesse (Seção IV-A1).
- **Segmentação:** Isolar os *pixels* da pele do motorista (Seção IV-A2).
- **Extração das características:** Extração de percentual de pele nas regiões em que habitualmente a mão/braço do motorista estaria ao usar celular e cálculo dos Momentos de HU [21] (Seção IV-A3).

Nas subseções seguintes as etapas do algoritmo posteriores á aquisição são explanadas.

1) Pré-processamento: Após a aquisição da imagem a etapa de pré-processamento é incumbida de localizar a região de interesse, ou seja, a face do motorista (maior face encontrada na cena). Para tanto, utilizou-se três detectores [1] baseado em *Haar-like-features* para extração de características (*features*) e *Adaboost* como classificador [22]. A maior área encontrada pelos três detectores é adotada pelo algoritmo como sendo a face do motorista. A Figura 3 exemplifica o resultado do processo de localização.

Ao conhecer a localização da face do motorista, pode-se extrair esta região preparando-se para os processos subsequentes. A região encontrada é acrescida em 20% sua largura, 10% para a esquerda e mais 10% para a direita da face. Este aumento faz-se necessário, pois muitas vezes, a região encontrada pelos classificadores é muito reduzida (somente deixa a face visível), tornando-se impossível a detecção da mão/braço. Na Figura 4 exemplifica-se o resultado obtido na etapa do pré-processamento.

[1]Arquivos Haar-like-features utilizados: haarcascade_frontalface_alt2.xml, haarcascade_profileface.xml e haarcascade_frontalface_default.xml.

Figura 3. Detecção da face do motorista

Figura 4. Extração da região da face

2) Segmentação: Para a segmentação do motorista utiliza dois passos principais: a localização da pele do motorista (ou parte dela) e a segmentação em tons de cinza. Precisa-se, nos processos, de três espaços de cores distintos: Tons de cinza (entre 0 e 255), *Hue Saturation Value* (HSV) [23] e o YCrCb [24].

Antes das conversões nos espaços de cores precisa-se utilizar um filtro de mediana [25], com um tamanho de janela de 3x3 *pixels*, para suavizar a imagem e minimizando possíveis ruídos de aquisição.

O passo seguinte é retirar 40 amostras de *pixels* situadas próximo à região central da imagem. Os valores absolutos de H, S, V, Cr e Cb são coletados e assim se conhece o alcance da segmentação da pele. Todos os pixels da imagem são filtrados baseados nestes intervalos. Resultado assim nas localizações dos pontos que atendam essas condições.

A segmentação em tons de cinza começa com a binarização da imagem através do método Otsu [26]. Esse método calcula um limiar automaticamente utilizando do histograma da imagem. Porém pode localizar partes da imagem que não façam parte da pele do motorista. Este problema é contornado pelos pontos da pele calculados pelos HSV e YCrCb que são agora dilatados condicionalmente ([27]) até mostrarem todas as partes encontradas pelo Otsu pertencentes a pele do motorista.

3) Características: Foram escolhidas duas características como forma de classificar se um motorista está ao telefone e são: Percentual da Mão (PM) e o Momento de Inércia (primeiro Momento de Hu [21]) (MI).

Figura 5. Resultado da Segmentação

Os PM são obtidos pela contagem dos pontos existentes em duas regiões da imagem que são mostradas na Figura 6. Basta fazer a contagem dos pontos existentes na Região 1 ($PT1$) e os pontos da Região 2 ($PT2$), dividindo pelo total de pontos total ($PontosTotal$). A fórmula do PM é expressa na Equação 8.

$$PM = \frac{PT1 + PT2}{PontosTotal} \tag{8}$$

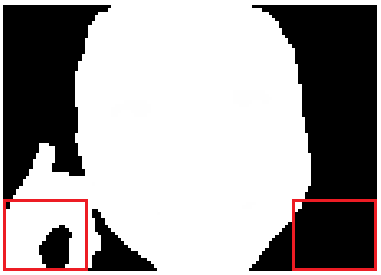

Figura 6. Regiões onde os pixels da mão/braço são contados

Já o MI é calculado usando o Momento de Hu. O momento de inércia mede a dispersão dos pontos na imagem.

V. Experimentos

Utilizou-se para testar o algoritmo de extração de características 58 imagens positivas (com celular) e 60 imagens negativas (sem celular). Todas foram tiradas com a pessoa em frente à câmera. Os resultados obtidos nos testes são expressos na Tabela 1.

Tabela I. Resultados obtidos nos testes do algoritmo de extração de características

Imagem	Característica	Média	Desvio Padrão	Variância
Com Celular (positivo)	PM	0,13348	0,05623	0,00316
	MI	0,00106	0,00045	0,00000
Sem Celular (negativo)	PM	0,03314	0,02347	0,00055
	MI	0,00098	0,0024	0,00000

O Desvio Padrão (DP) possui propriedades que dizem: 68% dos valores encontram-se a uma distância média inferior a um DP, 95% dos valores encontram-se a até duas vezes o DP e 99,7% dos valores encontram-se a até três vezes o DP. Analisando a característica PM das amostras nota-se que estão a uma distância entre um e dois DP, com isso, pode-se esperar que a acurácia do sistema esteja entre 68% e 95% para esta característica.

Realizaram-se experimentos com SVM e MLP para medir a acurácia do sistema, bem como, a técnica de classificação mais adequada. Utilizou-se em todos os testes o mesmo conjunto de características. Empregou-se *cross-validation* (validação cruzada) [28] criando 10 conjuntos de dados a partir do conjunto inicial.

A SVM testou-se com os *kernels*: Linear, Polinomial, RBF, e Sigmoide. De todos os *kernels* o que se apresenta como maior acerto é o Sigmoide que atingiu a taxa de 94,09%, muito próximo do teto da acurácia estatisticamente esperada de 95% para a característica PM. A Tabela 2 mostram-se os resultados dos testes, parâmetros utilizados e a acurácia de cada *kernel*. Na Figura 7 pode-se visualizar a distribuição dos dados (características) em teste realizado com *kernel* Sigmoide. Nela pode-se ver a linha divisória das duas classes criadas pelo SVM.

Tabela II. Acurácia dos Kernels SVM e parâmetros utilizados

Kernel	Parâmetros Extras	Acurácia
Linear	-	92,42%
Polinomial	$degree = 10$ $\gamma = 0,1$ $coef0 = 0,5$	93,25%
RBF	$\gamma = 25$	93,25%
Sigmoide	$\gamma = 25$ $coef0 = 0,5$	94,09%

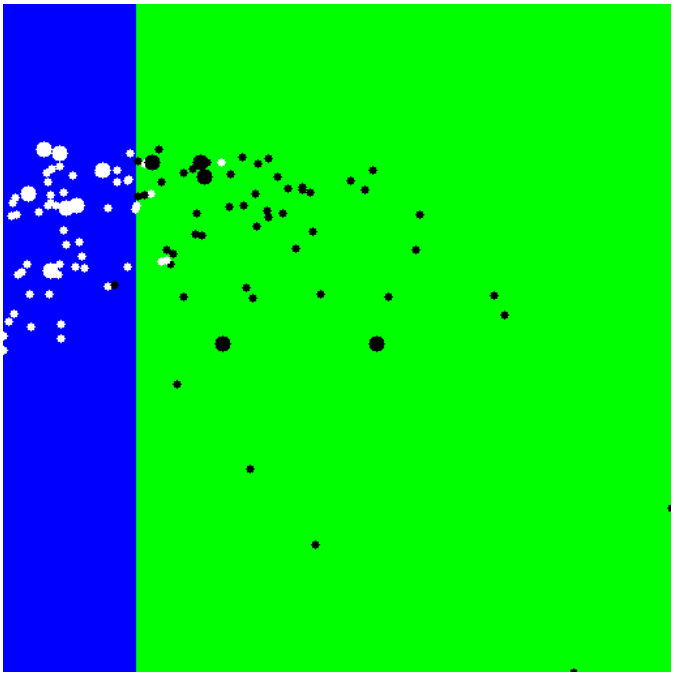

Figura 7. Dispersão dos dados em um conjunto de testes do SVM Sigmoide. Onde a altura da imagem representa os valores da característica MI e comprimento do PM já normalizados. Os círculos menores são os dados que compõem o treinamento, já os de maior diâmetro são dados de teste do sistema. Os dados em branco são provenientes de imagens rotuladas sem celular e dados em preto de imagens com celular. A região em verde indica a área em que a SVM considerará as características como sendo com celular, já a área em azul mostra-se a região considerada como sem celular.

Os experimentos realizados com a rede MLP utilizaram-se duas funções de ativação: uma Gaussiana e outra Sigmoide. Utilizou-se para isso uma rede de dois neurônios na camada de entrada, dois na camada oculta e um neurônio na camada de

saída. Em ambos os casos atingiu-se a acurácia de 91,59%. Nos testes utilizou-se *backpropagation* como forma de treinamento. Empregou-se o limite de 1000 gerações e aborta-se o treinamento antes de atingir este número se a acurácia baixar de $0,000001\%$. Na Tabela 3 mostram-se os resultados dos testes e parâmetros empregados. Vale destacar que os parâmetros bp_dw_scale e bp_moment_scale são, respectivamente, os coeficientes de gradiente de peso e de diferença entre os pesos das duas iterações anteriores. Esses são parâmetros utilizados no *backpropagation*.

Tabela III. ACURÁCIA DO MLP CONFORME FUNÇÃO DE ATIVAÇÃO E PARÂMETROS UTILIZADOS

Função de Ativação	Parâmetros Backpropagation	Acurácia
Gaussiana	$bp\_dw\_scale = 0,1$ $bp\_moment\_scale = 0,1$ $\alpha = 1$ $\beta = 1$	91,59%
Sigmoide	$bp\_dw\_scale = 0,1$ $bp\_moment\_scale = 0,1$ $\alpha = 1$ $\beta = 1$	91,59%

O conjunto SVM e Sigmoide foi o que obteve melhor acurácia. Realizou-se ainda um último experimento, utilizando-se para isso apenas a característica PM. Os resultados obtidos continuaram sendo os mesmo. Deduz-se então, que a característica MI é irrelevante para acurácia do sistema.

VI. CONCLUSÃO

Neste artigo foi apresentado um algoritmo que permite a extração de características de uma imagem possibilitando detectar a utilização do celular, pelo motorista, em um automóvel. Testaram-se os classificadores SVM e redes MLP como candidatas a resolver o problema. O sistema de classificação que se demonstrou ser o mais vantajoso foi o SVM com *kernel* Sigmoide que atingiu a taxa de acerto de 94,09%. Toda a detecção baseia-se no motorista de frente para a câmera (frontal).

No decorrer dos testes notou-se que a característica MI não influenciava na acurácia do sistema para as imagens de teste. Por isso, torna-se desnecessária a sua utilização.

Como forma de melhorar a taxa de acerto do algoritmo de detecção da pele do motorista, podem-se utilizar outras formas de segmentação em tons de cinza, que se adeque mais especificamente ao meio onde será utilizado.

REFERÊNCIAS

[1] M. Regan, J. Lee, and K. Young, *Driver distraction: Theory, effects, and mitigation*. CRC, 2008.

[2] M. Peissner, V. Doebler, and F. Metze, "Can voice interaction help reducing the level of distraction and prevent accidents?" 2011.

[3] A. Balbinot, M. Zaro, and M. Timm, "Funções psicológicas e cognitivas presentes no ato de dirigir e sua importância para os motoristas no trânsito," *Ciências e Cognição/Science and Cognition*, vol. 16, no. 2, 2011.

[4] T. Vanderbilt, "Porque dirigimos assim? e o que isso diz sobre nós, ed," *Campus, Rio de Janeiro*, 2009.

[5] A. SALVADOR, "A culpa foi do celular," 2011.

[6] IIHS, "Phoning while driving," *IIHS Status Report*, vol. 44, no. 9, pp. 1–8, 2009.

[7] NSC, "Understanding the distracted brain: Why driving while using hands-free cell phones is risky behavior," *National Safety Council - March 2010*, pp. 1–22, 2010.

[8] H. Veeraraghavan, N. Bird, S. Atev, and N. Papanikolopoulos, "Classifiers for driver activity monitoring," *Transportation Research Part C: Emerging Technologies*, vol. 15, no. 1, pp. 51–67, 2007.

[9] J. Yang, S. Sidhom, G. Chandrasekaran, T. Vu, H. Liu, N. Cecan, Y. Chen, M. Gruteser, and R. Martin, "Detecting driver phone use leveraging car speakers," in *Proceedings of the 17th annual international conference on Mobile computing and networking*. ACM, 2011, pp. 97–108.

[10] M. Watkins, I. Amaya, P. Keller, M. Hughes, and E. Beck, "Autonomous detection of distracted driving by cell phone," in *Intelligent Transportation Systems (ITSC), 2011 14th International IEEE Conference on*. IEEE, 2011, pp. 1960–1965.

[11] I. J. G. Enriquez, M. N. I. Bonilla, and J. M. R. Cortes, "Segmentacion de rostro por color de la piel aplicado a deteccion de somnolencia en el conductor."

[12] V. Vapnik, *The nature of statistical learning theory*. springer, 1995.

[13] P. Wagner, "Machine learning with opencv2," 2012.

[14] D. Meyer, "Support vector machines," *Porting R to Darwin/X11 and Mac OS X*, p. 23, 2001.

[15] I. Stanimirova, B. Üstün, T. Cajka, K. Riddelova, J. Hajslova, L. Buydens, and B. Walczak, "Tracing the geographical origin of honeys based on volatile compounds profiles assessment using pattern recognition techniques," *Food Chemistry*, vol. 118, no. 1, pp. 171–176, 2010.

[16] D. Kriesel, *A Brief Introduction to Neural Networks*, 2007, available at http://www.dkriesel.com.

[17] L. d. Silva, "Uso de redes neurais artificiais perceptron na previsão de consumo de energia elétrica: Quanto é previsível?" 2008.

[18] H. Nørgaard-Nielsen, "Foreground removal from wmap 5 yr temperature maps using an mlp neural network," *Astronomy and Astrophysics*, vol. 520, 2010.

[19] P. Werbos, *The roots of backpropagation: from ordered derivatives to neural networks and political forecasting*. Wiley-Interscience, 1994.

[20] F. Drews, M. Pasupathi, and D. Strayer, "Passenger and cell phone conversations in simulated driving." *Journal of Experimental Psychology: Applied*, vol. 14, no. 4, p. 392, 2008.

[21] M. Hu, "Visual pattern recognition by moment invariants," *Information Theory, IRE Transactions on*, vol. 8, no. 2, pp. 179–187, 1962.

[22] P. Viola and M. Jones, "Robust real-time object detection," *International Journal of Computer Vision*, vol. 57, no. 2, pp. 137–154, 2001.

[23] L. Brys, "Página dinâmica para aprendizado do sensoriamento remoto," 2008.

[24] F. Dadgostar and A. Sarrafzadeh, "An adaptive real-time skin detector based on hue thresholding: A comparison on two motion tracking methods," *Pattern Recognition Letters*, vol. 27, no. 12, pp. 1342–1352, 2006.

[25] H. Du and T. To, "Hand gesture recognition using kinect," 2011.

[26] B. Morse, "Lecture 4: Thresholding," *Brigham Young University, Utah*, 2000.

[27] E. Dougherty and R. Lotufo, *Hands-on morphological image processing*. Society of Photo Optical, 2003, vol. 59.

[28] R. Kohavi *et al.*, "A study of cross-validation and bootstrap for accuracy estimation and model selection," in *International joint Conference on artificial intelligence*, vol. 14. Lawrence Erlbaum Associates Ltd, 1995, pp. 1137–1145.