

IDENTIFICAÇÃO E RASTREAMENTO DE PESSOAS POR MEIO DE IMAGENS DIGITAIS CAPTURADAS A PARTIR DE CÂMERAS DE VIDEO

IDENTIFICATION AND TRACKING PEOPLE BY MEANS OF DIGITAL IMAGES CAPTURED FROM VIDEO CAMERAS

Lucas do Vale Manuel¹, Francisco Assis da Silva¹, Mário Augusto Pazoti¹, Danillo Roberto Pereira¹, Leandro Luiz de Almeida¹, Almir Olivette Artero²

¹Faculdade de Informática – FIPP, Universidade do Oeste Paulista – UNOESTE
e-mail: lucasdvmanuel@gmail.com, {chico, mario, danilopereria, llalmeida}@unoeste.br

²Faculdade de Ciências e Tecnologia – FCT, Universidade Estadual Paulista – Unesp
e-mail: almir@fct.unesp.br

RESUMO - A utilização de visão computacional desempenha um papel importante para fins de segurança, hoje em dia é amplamente utilizada e a cada dia vem ganhando novos recursos devido ao avanço tecnológico. A grande maioria dos sistemas de segurança atuais se baseia em um monitoramento onde a análise é feita pela interpretação do usuário. Para melhorar esse processo, têm sido propostos modelos de análise e reconhecimento de características de pessoas, tais como altura, cor da pele, reconhecimento facial, dentre outros, a fim de proporcionar a busca por reconhecimentos mais eficientes. Neste trabalho são utilizados diferentes algoritmos como HOG e SVM para identificação e rastreamento de pessoas, os espaços de cores RGB, HSV e YCrCb, são combinados e utilizados para a identificação e extração de características como tipo e cor de roupa de cada pessoa, para acompanhamento ao longo do vídeo.

Palavras-chave: HOG; SVM; Rastreamento de Pessoas; Espaço de Cores; Detecção de Pele.

ABSTRACT - The use of computational vision plays an important role for security purposes, nowadays it is largely used and, day by day, is earning new resources due to the technological progress. The majority of current security systems are based in a monitoring where the analysis is made by the user's interpretation. To improve this process, it has been proposed models of analysis and recognition of personal characteristics, like height, skin tone, facial recognition, among others, in order to provide the search for recognitions more efficient. In this work, there are used different algorithms like HOG and SVM for identification and tracking of people, color spaces RGB, HSV and YCrCb, that are combined and used for the identification and extraction of characteristics like the type and color of clothes of each person for monitoring along the video.

Keywords: HOG; SVM; Tracking People; Color Space; Skin Detection.

Recebido em: 10/03/2015
Revisado em: 03/09/2015
Aprovado em: 05/11/2015

1 INTRODUÇÃO

O rastreamento de pessoas em locais públicos é uma tarefa complexa e apresenta muitos desafios de visão computacional a serem resolvidos, como mudanças de iluminação do ambiente; velocidade com que as pessoas se locomovem; oclusão entre pessoas; mudanças bruscas na trajetória e as possíveis semelhanças entre as pessoas. Aplicações para rastreamento de jogadores de futebol, por exemplo, enfrentam

problemas com a iluminação artificial e fatores climáticos em estádios sem cobertura (TRUZZI; WERNECK; COSTA, 2010).

Nos sistemas de monitoramentos convencionais, feito por pessoas, a tarefa de observar um monitor por horas representa uma tarefa repetitiva e cansativa, o dificulta que essas pessoas consigam manter a atenção exclusiva por muito tempo, ocasionando grande probabilidade de erros (ROSA et al., 2012). Na maioria das

vezes, o monitoramento ainda é realizado por seres humanos, isso ocorre pelo fato de não existir um monitoramento eletrônico totalmente eficiente e também pela dificuldade de se conseguir desenvolver um sistema para essa tarefa, o que representa um grande desafio computacional.

Apesar das dificuldades encontradas para a identificação e rastreamento de pessoas, segundo Amaral e Tozzi (2013) o aumento do poder de processamento dos computadores têm motivado o uso de câmeras de monitoramento para tarefas de identificação e rastreamento de pessoas.

Este trabalho propõe, por meio da combinação de diferentes algoritmos, o desenvolvimento de um sistema de identificação e rastreamento de pessoas que auxilie a área de segurança. Busca-se identificar uma pessoa em um ambiente público, rastrear essa pessoa em sua trajetória, e para que isso possa ser realizado, são utilizadas características de roupas e pele dessa pessoa.

As demais seções deste trabalho estão organizadas da seguinte maneira: a Seção 2 apresenta trabalhos relacionados ao tema abordado; a Seção 3 apresenta os conceitos para a construção do HOG; a Seção 4 mostra as diferentes estratégias adotadas para detecção da pessoa humana; a Seção 5 apresenta a metodologia utilizada para a realização deste trabalho; a Seção 6 exibe os experimentos realizados e os resultados obtidos com a pesquisa; na Seção 7 são apresentadas as conclusões finais e trabalhos futuros.

2 TRABALHOS RELACIONADOS

No trabalho de Suliman, Cruceu e Moldoveanu (2010) foi desenvolvido um sistema de rastreamento em vídeos de segurança, com o intuito de prover uma identificação precisa das pessoas que passam pelas imagens, apresentando um bom desempenho nos processos de identificação e rastreamento. No desenvolvimento do trabalho foi utilizado o método de fluxo óptico Horn-Schunk para a identificação de movimentos nas imagens, e foi combinado com o algoritmo filtro de Kalman para detectar e rastrear as pessoas presentes nos *frames*. Nos experimentos, os autores utilizaram um vídeo com resolução de 240 x 320 pixels e verificaram que não era possível se obter os dois resultados esperados ao mesmo tempo (uma boa identificação com um bom desempenho). Uma

boa identificação torna-se custosa computacionalmente, já um bom desempenho acarreta em uma identificação comprometida.

Dalal e Triggs (2005) escolheram, através de experimentos, a combinação de SVM (*Support Vector Machine*) com o algoritmo HOG (*Histogram of Oriented Gradient*) para identificação e rastreamento de pessoas. O SVM por ser um algoritmo de aprendizagem supervisionada, foi utilizado para o treinamento na identificação de uma pessoa e, combinando com o HOG possibilitou ser utilizado para rastreamento. Nos experimentos foram utilizados os bancos de imagens MIT e INRIA, totalizando mais de 2500 imagens para treino e testes.

No trabalho de Sousa (2013) foi proposto um sistema para dispositivos móveis da plataforma Android, onde localmente eram processadas as imagens para identificação de faixas de pedestres e pessoas. O objetivo foi prover auxílio na locomoção segura dos deficientes visuais pelas ruas da cidade. Foi utilizado o espaço de cor YUV para que o algoritmo funcionasse em ambientes noturnos. A imagem é convertida em escalas de cinza para encontrar as faixas de pedestres, e em seguida são identificadas as pessoas que passam nessas faixas. No desenvolvimento do sistema, o autor utilizou o algoritmo HOG e o algoritmo SVM. Nos experimentos, dos 64 cenários avaliados, foram identificados 55 deles, utilizados imagens com resoluções de 480 x 640 e 240 x 320.

Truzzi, Werneck e Costa (2010) apresentou uma solução para detecção e rastreamento de pessoas, caminhando ou correndo, utilizando vídeos capturados por câmeras com posicionamento fixo. Os autores buscavam prover a identificação e rastreamento de pessoas, independente da distância e velocidade de deslocamento. Os autores utilizaram a subtração de fundo para identificar as pessoas presentes nas imagens e o filtro de Kalman para o rastreamento.

No trabalho de Amaral e Tozzi (2013) foi proposto um modelo para detecção e rastreamento de pessoas ao longo do vídeo. A ideia foi identificar e rastrear as pessoas tratando as oclusões das pessoas através separações das cabeças. Foi utilizada a subtração do fundo para localizar as pessoas presentes nas imagens, um algoritmo de identificação de cabeças para corrigir os problemas de oclusão e o filtro de Kalman para o rastreamento.

3 REFERENCIAL TEÓRICO

Nessa seção são apresentados os conceitos dos algoritmos usados neste trabalho. Inicialmente é apresentado o algoritmo HOG (*Histogram of Oriented Gradients*) juntamente com SMV (*Support Vector Machine*) para detecção de pessoas, na sequência é abordada a detecção de pele humana e, por fim, os espaços de cores RGB, HSV e YCbCr.

3.1 *Histogram of Oriented Gradients*

O Algoritmo HOG é uma técnica utilizada para detecção e extração de características de uma imagem, ela pode ser combinada com outros algoritmos para classificação destas características obtidas. Segundo Sousa (2013), o

método HOG é muito utilizado para o reconhecimento de pedestres e obtenção de suas características. HOG é utilizado em conjunto com SVM, que é um classificador para essas características. Sousa (2013) ainda complementa que o algoritmo HOG, SIFT e outros algoritmos são utilizados para extrair características da imagem, porém, o que a difere dos demais é que o HOG leva em consideração a imagem como um todo, já os demais algoritmos citados utilizam como base pontos específicos para a extração de características.

No trabalho de Dalal e Triggs (2005) foi proposto um modelo para a construção do HOG conforme as fases mostradas na Figura 1.

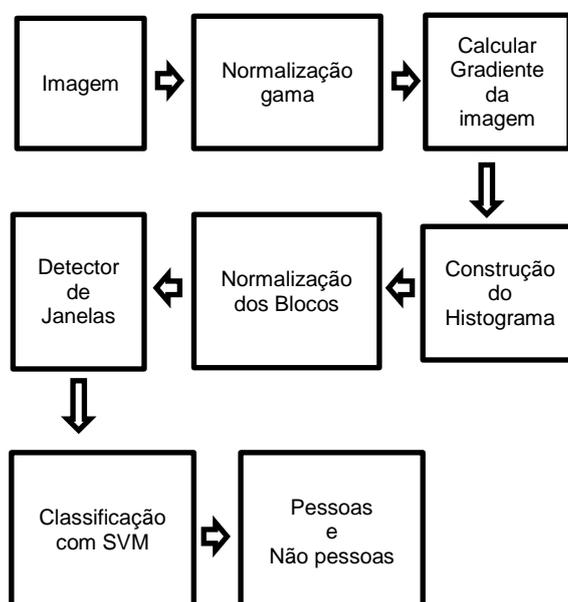


Figura 1. Modelo com sequência das fases de funcionamento do processo para construção do HOG. Fonte: (DALAL; TRIGGS, 2005).

As subseções seguintes detalham as fases de funcionamento do processo para a construção do HOG mostrado na Figura 1.

3.1.1 Normalização Gama

Nessa fase, a normalização gama é responsável pelo aumento no contraste das imagens em tons de cinza, no espaço de cor RGB ou no espaço de cor LAB, ajustando a iluminação das imagens para se obter um pequeno ganho de desempenho.

3.1.2 Calcular Gradiente da imagem

Nessa fase é realizado o cálculo do gradiente de cada pixel. Dalal e Triggs (2005) mostrou que o gradiente poderia ser calculado

através da derivada de cada pixel (Equação 1), onde é observado o valor do pixel superior subtraído do inferior, posteriormente o pixel da direita subtraído do pixel da esquerda. Com esses valores torna-se possível encontrar a magnitude do pixel (Equação 2) e a sua orientação (Equação 3).

Derivada do pixel

	123	
86		124
	85	

$$123 - 85 = 38 \quad (1)$$

$$124 - 86 = 38$$

$$|\nabla f| = \sqrt{(38)^2 + (38)^2} = 53.74 \quad (2)$$

$$\theta = \arctan\left(\frac{S_y}{S_x}\right) \quad (3)$$

O cálculo do gradiente é realizado em toda a imagem, porém, ela é subdividida em células com uma dimensão de 8x8 pixels. Cada célula possui um vetor para armazenar os 64 pares de valores correspondentes ao cálculo do gradiente.

3.1.3 Construção do Histograma

Após o cálculo do gradiente ter sido realizado, Dalal e Triggs (2005) construíram um histograma, um para cada célula, a partir do vetor resultado de 64 posições. Para cada par de valores do vetor é calculado o ângulo de orientação do gradiente do pixel usando a Equação 3. Os ângulos dos pixels são distribuídos nos *bins* do histograma (Figura 2).

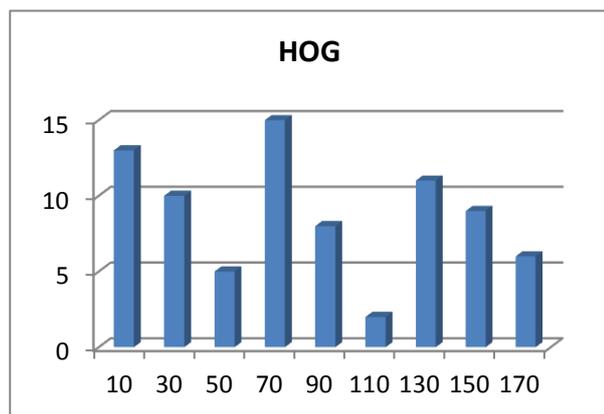


Figura 2. Exemplo de Histograma com as orientações.

Um histograma é representado por 9 barras (*bins*), e pertence a uma faixa de 0 a 180 graus e o intervalo entre cada barra é de 20 graus. Para a atribuição de valores em cada barra é considerado o valor do ângulo calculado para cada pixel. Esse valor então é dividido

proporcionalmente à faixa de valor em que seu ângulo se enquadra. A partir dos valores utilizados como exemplo nas Equações 1 e 2, tem-se representado na Figura 3 um exemplo de divisão para as barras do histograma.

Utilizando o valor resultante da derivada (Equação 1) e a orientação (Equação 3) foi obtido um ângulo de $\theta = 85^\circ$ (Figura 3).

$$\theta = \arctan\left(\frac{38}{38}\right)$$

$$\theta = 85^\circ$$

Então $70^\circ \leq \theta < 90^\circ$

Figura 3. Exemplo de cálculo da orientação de orientação do gradiente para divisão nas barras do histograma.

O ângulo θ é de 15° em relação à barra de 70° e 5° , em relação a de 90° , então a distribuição do valor para cada barra será da seguinte forma:

A barra de 90 recebe $15/20 = 3/4$ do valor, e a barra 70 recebe $5/20 = 1/4$ do valor. A barra de 90 recebe o maior valor por θ estar mais próximo.

3.1.4 Normalização dos Blocos

Os blocos são uma parcela da imagem compostas por 4 células cada, os blocos servem para mapear a imagem e através deles normalizar e varrer toda a imagem a procura de pessoas a serem identificadas. Na Figura 4 é mostrada a representação de um bloco.



Figura 4. Exemplo de um bloco em uma imagem em escalas de cinza.

Dalal e Triggs (2005) utilizaram em seus experimentos diferentes fórmulas para normalização dos blocos a fim de comparar o

desempenho de cada uma das fórmulas. As fórmulas utilizadas foram a fórmula L1-norm (Equação 4), fórmula L1-sqrt (Equação 5), fórmula L2-norm (Equação 6).

$$f = \frac{v}{(\|v\|_1 + e)} \quad (4)$$

$$f = \sqrt{\frac{v}{(\|v\|_1 + e)}} \quad (5)$$

$$f = \frac{v}{\sqrt{\|v\|_2^2 + e^2}} \quad (6)$$

A normalização dos blocos é feita de forma a sobrepor 50% de outro bloco e cada bloco terá sua normalização calculada separadamente.

3.1.5 Detector de Janelas

Criados todos os blocos, é construída a janela de detecção HOG. A janela HOG tem o tamanho de 64 por 128 pixels, é gerada a partir da junção de todos os blocos (2x2 células), células (8x8 pixels) e histogramas (9 bins por histograma) (DALAL; TRIGGS, 2005).

O detector de janela possibilita a varredura da imagem em um determinado sentido. O deslocamento segue da esquerda para a direita e da parte superior para parte inferior da imagem, buscando encontrar todas as pessoas presentes nela.

3.2 Detecção de Pele Humana

No trabalho Souza e Marana (2010) foi proposta a utilização do reconhecimento de características de altura e cor de pele para identificar pessoas. Para o reconhecimento das características, foram utilizadas faixas de cores do espaço RGB para identificar pele humana, diferença da imagem de *background* com a imagem de *foreground* para identificar onde a pessoa se encontra na cena, e calcular a altura das pessoas utilizando como referência a altura da porta, já conhecida, presente na cena.

Em busca de aprimorar os modelos de cores existentes para identificar pele, See et al. (2006) propuseram um modelo em que foi adicionado o canal *Hue* do espaço de cor HSV em um modelo já existente. O modelo existente é composto pelos espaços de cores RGB e YCbCR, e ao combinar com o novo canal, foi possível diminuir a frequência de erros e aumentar a taxa de acertos. Esse modelo de cores foi escolhido para ser utilizado neste trabalho para a identificação de pele humana.

3.3 Espaço de cor RGB, HSV e YCbCR

O espaço de cor RGB é composto de três cores primitivas, *Red* (R), *Green* (G) e *Blue* (B), que combinadas torna-se possível visualizar a grande maioria das cores (ENNEHAR; BRAHIM; HICHAM, 2010).

Segundo Rewar e Lenka (2013) o RGB é um modelo de cores básico e que todos os outros modelos são derivados dele. É utilizado para o processamento e armazenamento de imagens digitais, porém, por ser um modelo sensível à luz, dificulta na distinção das regiões coloridas. Na Figura 5 é apresentado o modelo de cor RGB.

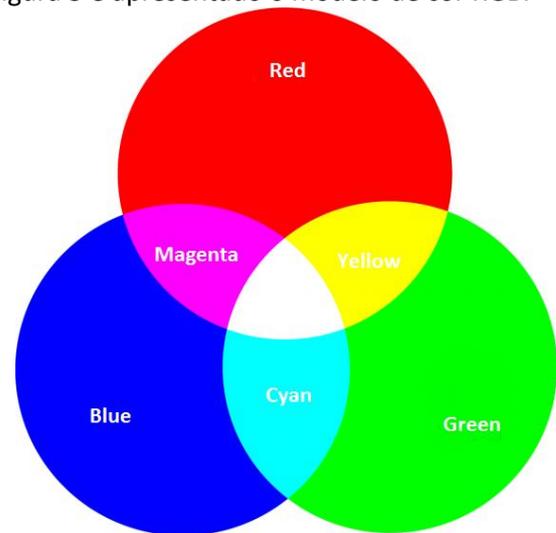


Figura 5. Modelo de cor RGB.

Fonte: (REWAR; LENKA, 2013).

O espaço de cor HSV contém três canais de cores, *Hue* (H) é o canal que define a percepção de cor, *Saturation* (S) que indica o quão pura é a cor, e *Value* (V) que descreve o quanto de brilho as imagens contém (AL-TAIRI et al., 2014) (REWAR; LENKA, 2013). Na Figura 6 é apresentado o modelo de cor HSV.

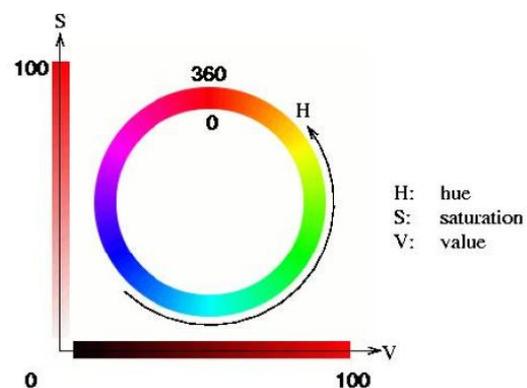


Figura 6. Modelo de cor HSV.

Fonte: (ENNEHAR; BRAHIM; HICHAM, 2010).

O espaço da cor YCbCr é composto pelo canal Y, responsável pela especificação da luminância, Canal Cb e Cr indicando o espaço de cores. O YCbCr é um canal modelo independente da luminância comparado ao modelo RGB (REWAR; LENKA, 2013).

Ennehar, Brahim e Hichan (2010) mostram que o YCbCr é um modelo usado em vídeos digitais, tanto para a transmissão de cores quanto para a compressão de imagens.

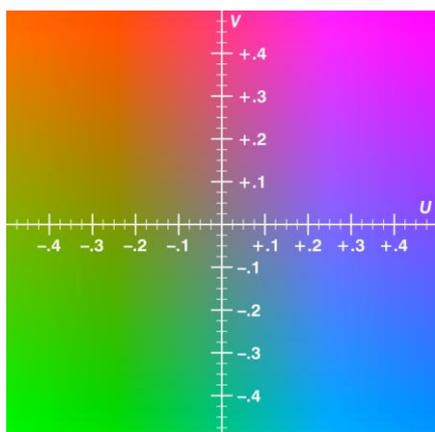


Figura 7. Modelo de cor YCbCr.
Fonte: (ENNEHAR; BRAHIM; HICHAM, 2010).

4 METODOLOGIA PROPOSTA

A metodologia proposta foi utilizada para a identificação, reconhecimento e rastreamento de pessoas. Todos os passos do processo estão ilustrados na Figura 8.

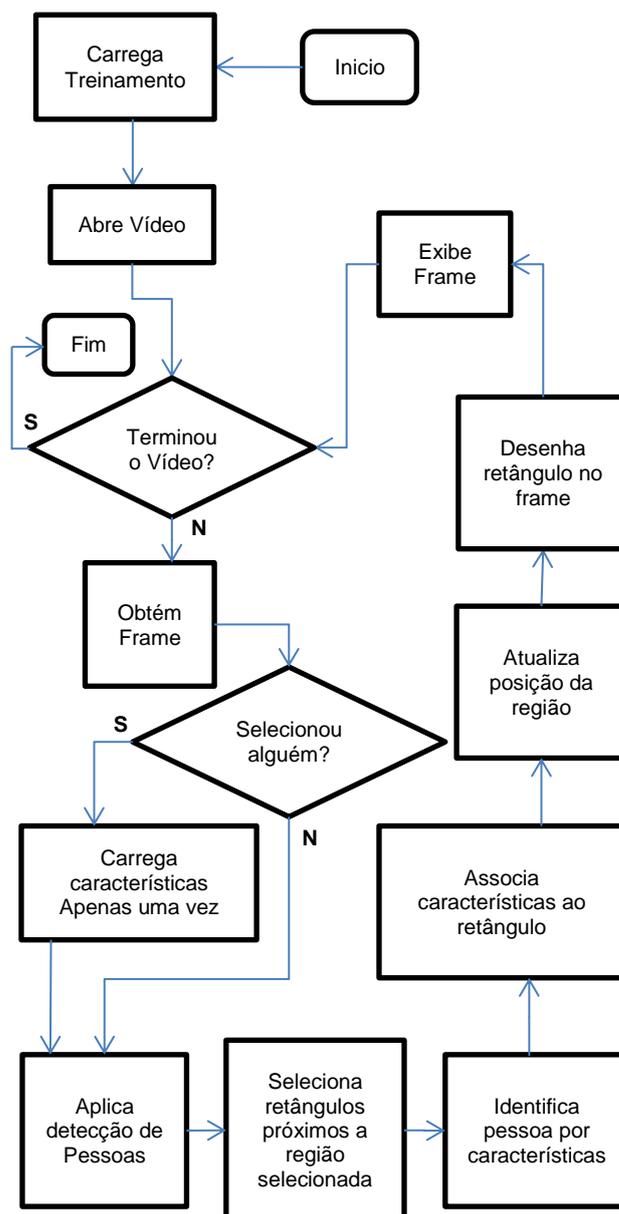


Figura 8. Metodologia proposta para a identificação, reconhecimento e rastreamento de pessoas.

A metodologia apresentada na Figura 8 descreve passo a passo o funcionamento dos processos de identificação e rastreamento das pessoas nos quadros do vídeo. Inicialmente, é carregado o arquivo de treinamento disponível na biblioteca de Visão Computacional OpenCV. Um exemplo de parte do arquivo de treinamento disponível na biblioteca OpenCV é apresentado na Figura 9.

Name	Value	Type
[0]	0.053593859	float
[1]	-0.14721455	float
[2]	-0.055321701	float
[3]	0.050773069	float
[4]	0.11547081	float
[5]	-0.042688038	float
[6]	0.046358339	float
[7]	-0.054681990	float
[8]	0.082320839	float
[9]	0.10424068	float
[10]	-0.022945181	float
[...]
[3770]	-0.0014609738	float
[3771]	0.14013411	float
[3772]	-0.0089618182	float
[3773]	-0.032502461	float
[3774]	0.0033863019	float
[3775]	0.0026477948	float
[3776]	0.033597320	float
[3777]	-0.024119910	float
[3778]	-0.042297289	float
[3779]	0.10666174	float
[3780]	-6.6657915	float

Figura 9. Exemplo de parte do arquivo de treinamento disponibilizado pela biblioteca OpenCV.

O arquivo de treinamento é composto por um conjunto de 3.781 elementos. As 3.780 posições iniciais são utilizadas em conjunto com 3.780 posições presentes na janela de detecção, e o último valor é utilizado para classificação da imagem juntamente com o *hit_threshold* (valor utilizado de limite de verificação) utilizado para estipular um intervalo válido de classificação.

As 3.780 posições presentes na janela de detecção são equivalentes aos *bins* de uma imagem de 68 x 128 pixels. Em uma imagem com essa dimensão tem-se 7 blocos na horizontal e 15 blocos na vertical, totalizando 105 blocos. Cada bloco contém 4 células, totalizando 420 células, e cada célula contém 9 *bins*, totalizando as 3.780 posições presentes na janela de detecção.

A classificação é feita durante a execução do programa após o vídeo ser carregado. É feita a somatória dos *bins* da janela com o vetor de treinamento, e caso o valor resultante esteja no intervalo determinado, a imagem é classificada como positiva, ou seja, possui uma pessoa nela. Em casos de oclusão não é possível detectar todas as pessoas presente no *frame*, pois não houve a implementação neste trabalho. Um exemplo de como é feita a classificação é apresentado na Figura 10.

Arquivo de Treinamento		Janela de Detecção		Somatória		
0	-0,05359	+	0	-0,14721	+=	-0,20081
1	0,147215	+	1	0,002648	+=	-0,05095
2	0,055322	+	2	0,115471	+=	0,119846
3	-0,05077	+	3	-0,0935	+=	-0,02443
4	-0,11547	+	4	-0,05359	+=	-0,1935
5	0,042688	+	5	0,053594	+=	-0,09721
6	-0,04636	+	6	-0,05759	+=	-0,20116
7	0,054068	+	7	0,104241	+=	-0,04286
8	-0,08232	+	8	-0,02359	+=	-0,14877
9	-0,10424	+	9	-0,00359	+=	-0,25661
10	0,022945	+	10	0,055094	+=	-0,17857
...		+
3770	0,001461	+	3770	0,147215	+=	-5,85132
3771	-0,14013	+	3771	-0,03959	+=	-6,03105
3772	0,008962	+	3772	0,033553	+=	-5,98854
3773	0,032502	+	3773	0,106662	+=	-5,84937
3774	-0,00339	+	3774	-0,01159	+=	-5,86435
3775	-0,00265	+	3775	-0,05335	+=	-5,92035
3776	-0,0336	+	3776	-0,0505	+=	-6,00445
3777	0,02412	+	3777	0,106662	+=	-5,87367
3778	0,042297	+	3778	0,055594	+=	-5,77578
3779	-0,10666	+	3779	-0,00309	+=	-5,88553

Figura 10. Exemplo da somatória do arquivo de treinamento com a janela de detecção.

Com o valor da somatória é então verificado se ele permanece ao intervalo para casos positivos através da seguinte condição:
 $-6,6657915 - hit_threshold > Somatória$ AND
 $-6,6657915 + hit_threshold < Somatória$

Com o arquivo de treinamento carregado, o sistema desenvolvido para identificação e rastreamento de pessoas inicia sua execução, até que o vídeo termine. A cada ciclo, obtém-se um novo *frame* do vídeo para ser tratado, porém é utilizado para análise um a cada cinco *frames*. A cada *frame* analisado é verificado se houve a seleção de algum local da imagem, essa verificação é feita através de variáveis de controle. A Figura 11 mostra a execução do sistema com a detecção de duas pessoas.

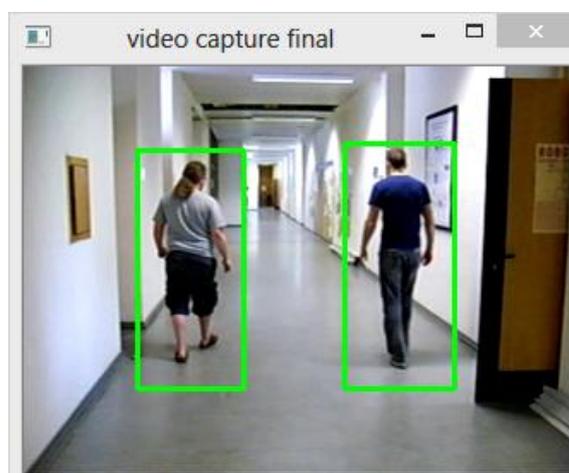


Figura 11. *Frame* do vídeo com duas pessoas detectadas.

Ao selecionar uma das pessoas presentes na Figura 11, são registradas as seguintes características pertencentes à pessoa: tipo roupa superior (manga curta ou manga longa), cor da roupa superior, tipo da roupa inferior (calça ou bermuda) e cor da roupa inferior.

No processo seguinte a imagem passa por diversos passos na tentativa de identificar pessoas presentes nela. Para a identificação de pessoas é utilizada a normalização gama, cálculo do gradiente da imagem, construção dos histogramas, normalização dos blocos, detector de janelas e por fim, sua classificação utilizando SVM. Todos esses passos estão descritos na Seção 3.

Com a identificação, encontram-se as pessoas, obtendo como resultado as posições dos retângulos que pertençam às pessoas existentes na imagem. Encontrado os retângulos, é realizada a seleção dos retângulos próximos à região selecionada, minimizando o trabalho de verificar todos os retângulos presentes no *frame*. Essa seleção é feita a partir da variável de controle da posição selecionada, e a partir dela verifica-se a existência de retângulos próximos a essa área (Figura 12).

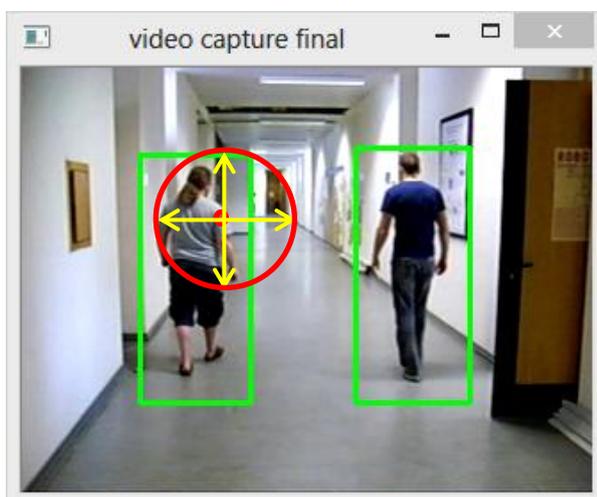


Figura 12. Exemplo de seleção de retângulos.

Selecionado os retângulos desejados, cada um deles terá sua extração e verificação feita separadamente. Para o processo de extração e verificação das características é necessária uma preparação das imagens para poderem ser utilizadas. O primeiro passo para essa preparação é a identificação da pele humana.

Para a identificação da pele humana foi utilizado o método proposto por See et al. (2006).

O método realiza uma varredura em cada pixel da imagem, verificando nos três espaços de cores, RGB, HSV e YCbCr os intervalos válidos que representam pele humana. O pixel que passa pela verificação de todos os filtros será mantido na imagem, caso contrário, será atribuído um pixel preto em substituição a ele. Para verificação do intervalo nos três espaços de cores são utilizadas as seguintes condições:

- HSV
 $H > 230$ or $H < 25$
- YCrCb
 $Cr \leq 1.5862 \times Cb + 20$ AND
 $Cr \geq 0.3448 \times Cb + 76.2069$ AND
 $Cr \geq -4.5652 \times Cb + 234.5652$ AND
 $Cr \leq -1.15 \times Cb + 301.75$ AND
 $Cr \leq -2.2857 \times Cb + 432.85$
- RGB
 $((R > 95) \text{ AND } (G > 40) \text{ AND } (B > 20) \text{ AND } (\max\{R,G,B\} - \min\{R,G,B\} > 15) \text{ AND } (|R - G| > 15) \text{ AND } (R > G) \text{ AND } (R > B))$
 OR
 $((R > 220) \text{ AND } (G > 210) \text{ AND } (B > 170) \text{ AND } (|R - G| \leq 15) \text{ AND } (R > B) \text{ AND } (G > B))$

Ao final da varredura da imagem, apenas pixels que representem a cor da pele permanecerão na imagem, os outros pixels são trocados para preto. O resultado é apresentado na Figura 13.



Figura 13. Exemplo do resultado de identificação de pele humana utilizando os espaços de cores RGB, HSV e YCbCr.

Com a imagem contendo apenas regiões com pixels próximos a tonalidade da pele humana, inicia-se uma varredura em busca dos pixels que pertencem ao intervalo desejado (pixels das regiões de pele) para trocá-los pela cor branca.

Em seguida é utilizado o algoritmo de Canny (CANNY, 1986) para detecção de contornos da imagem. O algoritmo Canny é composto de 4 passos, sendo que no primeiro, é aplicado um filtro gaussiano na imagem com máscara 5 x 5 para reduzir o ruído, conforme exemplo a seguir:

$$\frac{1}{273} \begin{bmatrix} 1 & 4 & 7 & 4 & 1 \\ 4 & 16 & 26 & 16 & 4 \\ 7 & 26 & 41 & 26 & 7 \\ 4 & 16 & 26 & 16 & 4 \\ 1 & 4 & 7 & 4 & 1 \end{bmatrix}$$

Posteriormente são utilizados dois *kernels* diferentes para encontrar as bordas presentes na imagem, uma para bordas na horizontal e outra para vertical:

-1	0	1
-2	0	2
-1	0	1

Horizontal

-1	-2	-1
0	0	0
1	2	1

Vertical

Como resultado, a Figura 14 mostra as duas imagens geradas pelos *kernels* e a imagem final com a detecção de todas as bordas.

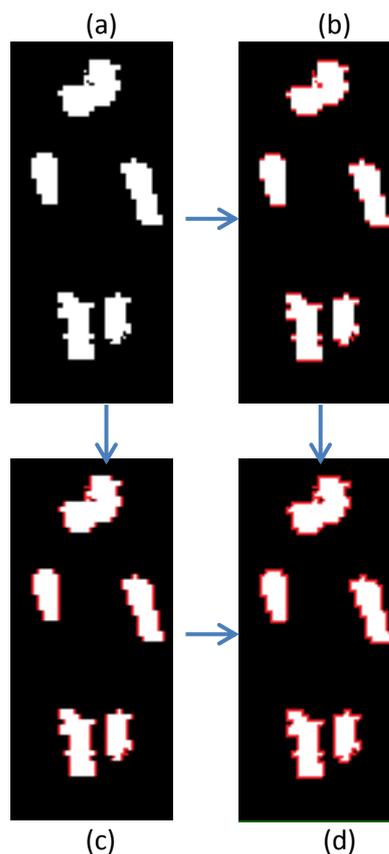


Figura 14. Exemplo do resultado da detecção de bordas horizontais (b) e verticais (c) a partir da imagem original em (a). Por fim, a imagem (d) mostra o resultado da junção da imagem (b) com a imagem (c).

Com a imagem resultante da detecção de bordas, são calculadas as intensidades de gradiente e a direção das bordas utilizando as seguintes fórmulas:

$$Edge_{Gradiente(G)} = \sqrt{G_x^2 + G_y^2} \quad (8)$$

$$Angle(\theta) = \tan^{-1}\left(\frac{G_y}{G_x}\right) \quad (9)$$

Os pixels que não representem borda são eliminados. É feita novamente uma varredura na imagem pixel a pixel verificando se ele representa o limite da borda, eliminando assim os pixels indesejados.

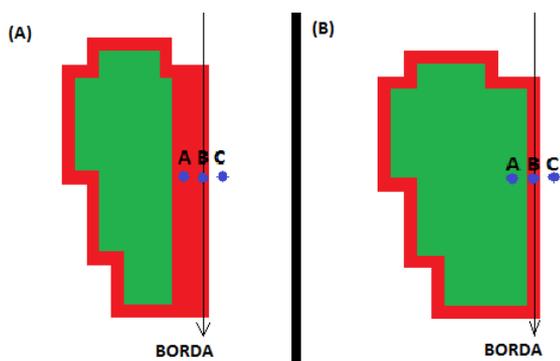


Figura 15. Exemplo de varredura na imagem, em que é mantida apenas a linha dada por B, representando o limite da borda.

E por fim, são verificados os pixels restantes para classificá-los como bordas e não bordas, através dos limites de *thresholding* máximo e mínimo.

Finalizando assim os passos de preparação da imagem. Os resultados são ilustrados na Figura 16.

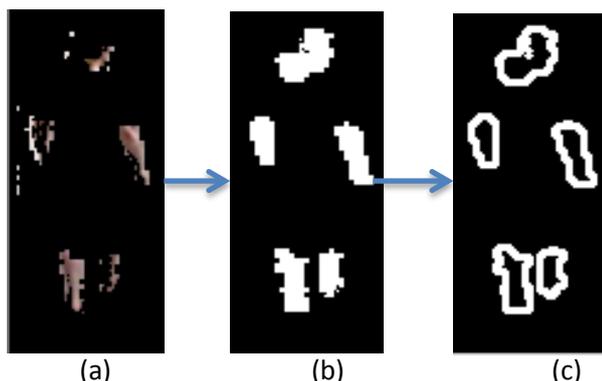


Figura 16. Exemplo do resultado dos métodos (a) detecção de pele, (b) região de pele substituída pela cor branca e (c) aplicação do algoritmo Canny.

Após a preparação da imagem, torna-se possível extrair a característica de tipo de roupa, que é realizada, inicialmente, utilizando o algoritmo de Border Following (algoritmo para extração das bordas em imagens binárias).

Ao aplicar o algoritmo Border Following, obtêm-se os conjuntos de vértices de cada região detectada. Cada conjunto contém as posições dos vértices das bordas necessárias para representar os membros da pessoa detectada: cabeça, braços e pernas (Figura 17).

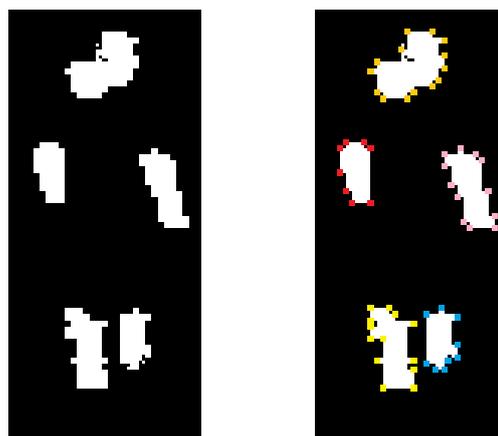


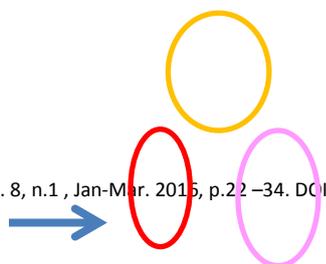
Figura 17. Resultado indicando cinco conjuntos diferentes de vértices.

A partir das posições de cada conjunto de vértices, calcula-se a dimensão da área que a borda abrange. Os conjuntos são utilizados para se verificar a qual região eles pertencem: parte inferior ou superior da imagem. Para indicar qual tipo de roupa que a pessoa está usando, faz-se necessário saber a área de cada conjunto.

O cálculo da área dos conjuntos das bordas é realizado através do teorema de Green (STEWART, 2008). Esse teorema utiliza a seguinte fórmula para o cálculo de área:

$$A = \oint_C xdy = - \oint_C ydx = \frac{1}{2} \oint_C (xdy - ydx) \quad (10)$$

Calculada a área da imagem, é possível estimar qual é o tipo de roupa utilizada pela pessoa. Dependendo do tamanho, a peça de roupa será classificada como curta ou longa, calça ou short (Figura 18).



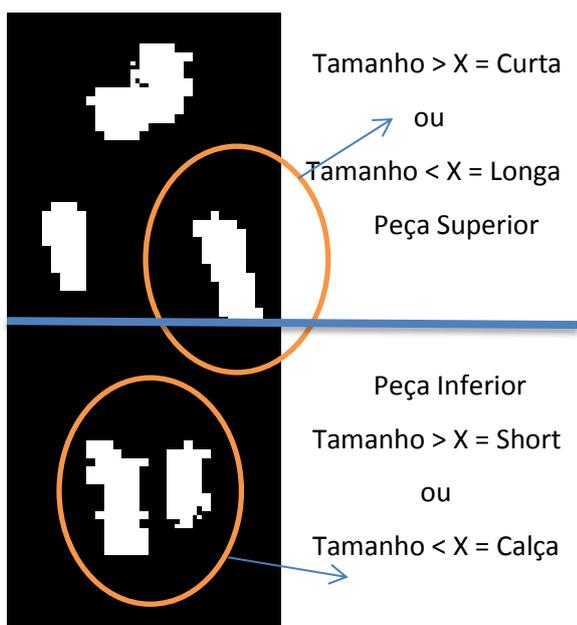


Figura 18. Exemplo verificação de qual tipos de roupas estão sendo usadas pela pessoa.

A próxima característica extraída é a cor das roupas. A imagem em RGB é convertida pra HSV utilizando as seguintes fórmulas de conversão:

$$V \leftarrow \max(R, G, B) \quad (11)$$

$$S \leftarrow \begin{cases} \frac{V - \min(R, G, B)}{v} & \text{if } V \neq 0 \\ 0 & \text{else} \end{cases} \quad (12)$$

$$H \leftarrow$$

$$\begin{cases} 60(G - B)/(V - \min(R, G, B)) & \text{if } V = R \\ 120 + 60(R - B)/(V - \min(R, G, B)) & \text{if } V = G \\ 240 + 60(R - G)/(V - \min(R, G, B)) & \text{if } V = B \end{cases} \quad (13)$$

Com a imagem convertida para HSV, serão extraídos dois pixels da imagem em posições fixas, um dos pixels representa a cor da peça superior da pessoa e o outro a peça inferior. Os valores do canal H dos dois pixels são armazenados em um vetor de características. Exemplo desse processo na Figura 19.



Figura 19. Exemplo extração de cores das peças de roupas.

As características das regiões de interesse (tipos e cores da roupa superior e inferior) são utilizadas para rastrear a pessoa selecionada. Na Figura 19, as características são comparadas. O retângulo que envolve a pessoa selecionada é atualizado *frame a frame*, indicando seu deslocamento durante a execução do vídeo. A Figura 20 mostra um *frame* contendo duas pessoas envolvidas por retângulos, um na cor vermelha (pessoa selecionada de interesse) e o outro na cor verde.

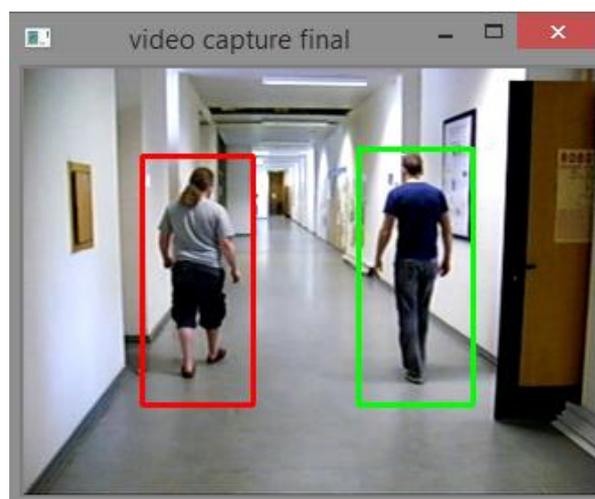


Figura 20. Exemplo de *frame* mostrando a pessoa selecionada (retângulo na cor vermelha) durante o decorrer do vídeo.

5 EXPERIMENTOS

Nos experimentos foram utilizados dois vídeos obtidos da internet com as seguintes propriedades: O primeiro vídeo tem 5 minutos de

duração, 25 *frames* por segundo, e resolução de 1920 x 1080 pixels (http://www.robots.ox.ac.uk/~lav/Research/Projects/2009bбенfold_headpose/Datasets/TownCentreXVID.avi); o segundo vídeo tem 15 segundos de duração, 25 quadros por segundo e resolução de 240 x 320 pixels (https://www.youtube.com/watch?v=8yMErr_zwQs&feature=youtu.be).

Os experimentos foram iniciados utilizando o primeiro vídeo, porém, durante os testes percebeu-se que a resolução alta do vídeo implica significativamente no desempenho do programa, o tempo de processamento nesta resolução foi acima de nove segundos por *frame*. Como meio de contornar esse problema, o vídeo foi reduzido para 1/4 do seu tamanho, mesmo assim os resultados não foram satisfatórios. Com a diminuição do vídeo, as pessoas ficaram menores, e isso fez com que o algoritmo não mais as identificasse.

Como alternativa foi utilizado o segundo vídeo. Por apresentar resolução menor que o primeiro, esse vídeo interfere significativamente para o desempenho dos testes. Com o novo vídeo, o tempo médio para o processamento de cada *frame* foi de 140 milissegundos. Nos primeiros experimentos estavam sendo verificados todos os *frames* do vídeo. Visto que o processo ainda tinha algumas limitações de processamento, foi utilizado um a cada 5 *frames*.

Foram utilizados no desenvolvimento da metodologia os algoritmos implementados pelo OpenCV (algoritmo HOG e SVM) para a identificação e rastreamento das pessoas, e implementação do algoritmos para combinação de 3 espaço de cores, RGB, HSV e YCrCb e identificar a cor da pele.

Durante os experimentos foram utilizadas duas variáveis configuráveis para testar o desempenho do algoritmo. As variáveis utilizadas foram:

Quantidade válida de características: nesse item é determinada a quantidade mínima de características para a identificação de uma pessoa, as características analisadas são tipo e cor da roupa superior e inferior.

Ajuste de posição: nesse item é determinada a área a ser verificada, buscando procurar encontrar pessoas nas proximidades da última detecção para o seu reconhecimento.

Durante o vídeo foram processados 78 frames, e em cada frame contêm 2 pessoas, totalizando 156 pessoas a serem identificadas.

$$78 * 2 = 156 \text{ pessoas} \quad (14)$$

Nos testes realizados, foram obtidos os seguintes resultados:

- Resultados de Identificação.

Frames	Pessoas
78	156

Identificações		Acertos (%)
Esperadas	156	100
Obtidas	142	91,025

- Resultados de Reconhecimento

Esperados	
Frames	78
Reconhecimento	78
Acertos (%)	100
Média (%)	100

Obtidas com ajuste posição = 10			
Características	4	>= 3	>= 2
Frames	78	78	78
Reconhecimento Médio	31,67	49,66	70,32
Acerto Médio (%)	40,16	63,67	90,16

Obtidas com ajuste posição = 5			
Características	4	>= 3	>= 2
Frames	78	78	78
Reconhecimento Médio	56,16	71,76	72,66
Acerto Médio (%)	72	92	93,15

6 CONSIDERAÇÕES FINAIS

Esse trabalho foi motivado pela dificuldade que usuários de circuito fechado de segurança têm em visualizar diversos ambientes monitorados por câmeras. A ausência de um sistema que os auxiliem na área da segurança é algo que vem ganhando importância a cada dia. OAcredita-se que o trabalho é relevante,

mostrando a possibilidades de utilizar novas combinações, que até então não haviam sido reunidas para essa finalidade. Diferentes algoritmos são utilizados para identificação e rastreamento de pessoas a partir da extração de suas características. Os resultados obtidos nos testes foram considerados satisfatórios, o que justifica ainda mais a relevância do trabalho.

Como trabalho futuro, pode-se utilizar como base a mesma metodologia proposta, e a partir dela aplicar o reconhecimento para um circuito de câmeras. Uma melhoria para o trabalho proposto, seria a otimização da metodologia, utilizando GPU ou CUDA como um acelerador para o processo.

REFERÊNCIAS

- AL-TAIRI, Z. G. et al. Skin segmentation using YUV and RGB color spaces. **Journal Of Information Processing Systems**, v.10, n.2, p.283-299, june 2014. <http://dx.doi.org/10.3745/JIPS.02.0002>
- AMARAL, W.; TOZZI, C. L. Tracking people through occlusions using a head detection method. In: WORKSHOP DE VISÃO COMPUTACIONAL (WVC 2013), 9. **Proceedings...** Rio de Janeiro/RJ: FGV, 2013.
- CANNY, J. A computational approach to edge detection. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 8, n. 6, p. 679-698, 1986. <http://dx.doi.org/10.1109/TPAMI.1986.4767851>
- DALAL, N.; TRIGGS, B. Histograms of oriented gradients for human detection. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2005). **Proceedings...** San Diego/USA, p. 886-893, v.1, 2005. <http://dx.doi.org/10.1109/cvpr.2005.177>
- ENNEHAR, B. C.; BRAHIM, O.; HICHAM, T. An appropriate color space to improve human skin detection. In: Laboratory of Automatics and Informatics of Guelma (LAIG 2010). **Proceedings...** Guelma/Algeria: Eight May 1945 University of Guelma, 2010.
- REWAR, E.; LENKA, S. K. Comparative analysis of skin color based models for face detection. **Signal & Image Processing: An International Journal**, v.4, n.2, april 2013. <http://dx.doi.org/10.5121/sipij.2013.4206>
- ROSA, A. L. B. et al. Sistema de rastreamento de objetos através de múltiplas câmeras. In: Seminário de Computação (SEMINCO 2012), 21. **Anais...** Blumenau/SC: FURB, 2012. p.28-37.
- SEE, J. et al. RGB-H-CbCr skin model for human face detection. In: MMU INTERNATIONAL SYMPOSIUM ON INFORMATION AND COMMUNICATION TECHNOLOGIES (M2USIC 2006). **Proceedings...** Malaysi, 2006.
- SOUSA, K. A. O. Uso de visão computacional em dispositivos móveis para auxílio à travessia de pedestres com deficiência visual. In: PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA (PPGEE 2013). **Anais...** São Paulo/SP: Universidade Presbiteriana Mackenzie, 2013.
- SOUZA, G. B.; MARANA, A. N. Reconhecimento de pessoas em imagens de vídeo utilizando características biométricas leves. In: WORKSHOP DE VISÃO COMPUTACIONAL (WVC 2010), 6. **Proceedings...** Presidente Prudente: Unesp, 2010.
- STEWART, J. **Multivariable calculus**. Belmont: Books/Cole, 2008. Disponível em: <https://books.google.com.br/books?id=K1LyXtIFDbkC&printsec=frontcover&hl=pt-BR&source=gbs_ge_summary_r&cad=0#v=onepage&q&f=false>. Acesso em: 30 abr. 2015.
- SULIMAN, C.; CRUCERU, C.; MOLDOVEANU, F. Kalman filter based tracking in an video surveillance system. In: International Conference on DEVELOPMENT AND APPLICATION SYSTEMS (DAS 2010), 10. **Proceedings...** Suceava/Romania: Universitatea Stean cel Mare Suceava, p. 54-58. <http://dx.doi.org/10.4316/aece.2010.02005>
- TRUZZI, F.; WERNECK, N.; COSTA, A. Rastreamento de pessoas em sequências de imagens capturadas por uma câmera estática. In: WORKSHOP DE VISÃO COMPUTACIONAL (WVC 2010), 5. **Proceedings...** Presidente Prudente: Unesp, 2010. p. 248-253.