

3 Subtração de Fundo

Este capítulo apresenta um estudo sobre algoritmos para a detecção de objetos em movimento em uma cena com fundo estático. Normalmente, estas cenas estão sob a influência de mudanças na iluminação e sujeitas à presença de sombras. O objetivo principal deste capítulo é estabelecer um algoritmo para detectar e segmentar uma mão em movimento sobre um fundo qualquer a partir de uma seqüência de imagens em cores, capturadas por uma câmera. Para isso, são estudados diversos algoritmos de subtração de fundo, com a finalidade de achar abordagens capazes de lidar com fundos arbitrários sujeitos a mudanças na iluminação e à presença de sombras.

A seção seguinte apresenta um breve resumo da técnica de segmentação baseada em subtração de fundo seguido de uma discussão sobre alguns dos problemas dos algoritmos encontrados na literatura para o problema proposto. Posteriormente são descritos dois algoritmos selecionados para serem implementados em função deles serem capazes de lidar com mudanças de iluminação e sombras. Finalmente os resultados destes dois algoritmos são apresentados, comparados e analisados. Desta análise resulta uma escolha em função do objetivo desta etapa de segmentação.

3.1. Segmentação por Subtração de Fundo

A capacidade de detectar objetos em movimento a partir de seqüências de vídeo é fundamental em muitos sistemas de visão computacional. Essa capacidade permite que os sistemas foquem a atenção nos objetos que estão em movimento e que possivelmente são uma peça crucial na execução da tarefa para a qual foram programados. Uma abordagem muito utilizada para a segmentação de objetos em movimento do fundo da cena é a *subtração de fundo*. A idéia da subtração de fundo é subtrair da imagem atual uma imagem de referência, a qual é adquirida e modelada a partir de um fundo estático durante um certo período de tempo, conhecido como *período de treinamento*. Assim, o fundo pode ser qualquer um, desde que permaneça razoavelmente estático.

A qualidade dos resultados obtidos pelo processo de subtração de fundo depende muito do modelo utilizado para a representação do fundo da cena. Uma abordagem simples e às vezes satisfatória consiste em calcular uma imagem de referência do fundo como sendo a média de um conjunto de imagens. Posteriormente, quando um objeto entra na cena, este causa uma divergência na imagem capturada com relação à imagem de referência. Assim, os pixels da imagem capturada correspondentes ao objeto são aqueles nos quais ocorrem mudanças significativas.

Desde que a técnica de subtração de fundo foi reconhecida como um método para detectar objetos em movimento, apareceram na literatura diversos modelos e abordagens para a implementação destes algoritmos. Alguns algoritmos exploram a diferença estatística e/ou probabilística de cor entre a imagem atual e a imagem de referência, a qual é treinada durante um período de tempo ou com base em um número determinado de imagens [14, 15, 6, 13, 2, 3]. Outros se baseiam na análise de movimento dos objetos [17], nas características das imagens em estéreo [45], em transformações logarítmicas [46] e no aprendizado por markov e bayesianas [20]. Existem ainda algumas abordagens híbridas (por exemplo, [17, 46, 5]).

No contexto da pesquisa desta dissertação, o algoritmo de segmentação de fundo foi utilizado para segmentar o objeto “MÃO” de um fundo estático qualquer sobre o qual ela está se movimentando.

3.2. Requisitos para o Algoritmo de Subtração de Fundo

O sistema de reconhecimento de gestos precisa processar as imagens capturadas em tempo real. Como a etapa de extração de fundo é apenas a parte inicial deste sistema, a eficiência medida em termos de tempo de processamento é um requisito importante.

Existem outros problemas que tornam o processo de segmentação, via subtração de fundo, bastante difícil. Qualquer mudança física na iluminação altera a imagem do fundo, o que torna o modelo inadequado e afeta o desempenho do algoritmo.

Embora muitos dos algoritmos disponíveis na literatura tenham desempenhos satisfatórios, são poucos os que conseguem lidar razoavelmente bem com as mudanças globais e locais da iluminação, tais como sombras e flutuações nas fontes de luz.

Também é importante considerar o fator de robustez e eficiência na detecção do objeto em movimento, característica que visa permitir um grau de detecção alto e uma segmentação que represente realmente a morfologia do objeto. No contexto desta pesquisa as medições foram feitas baseadas na qualidade de detecção e segmentação da morfologia da mão.

O próprio movimento dos objetos gera sombras e ocasiona mudanças na iluminação. Às vezes estas sombras são detectadas como sendo parte do objeto, afetando a eficácia do algoritmo. Os algoritmos de Wu & Jeng [46] e de Butler et al. [2], por exemplo, têm seu desempenho muito prejudicado pela presença de sombras. Na implementação que fizemos destes algoritmos as sombras foram detectadas como sendo parte do objeto e não encontramos uma alternativa para destacá-las, de modo que a morfologia da mão ficou totalmente deformada na segmentação. Como é afirmado pelos autores destes algoritmos, eles só são adequados para a extração de fundos que não contenham sombras do objeto.

Existem ainda os algoritmos que tratam das sombras, mas que se tornam instáveis na presença delas. Na nossa implementação dos algoritmos de Elgammal et al. [6] e de Hong & Woo [13] muitos pontos da sombra do objeto foram considerados como sendo o objeto. Ajustando os limiares destes algoritmos (valores de *threshold*) pode-se melhorar esta situação, no entanto esse procedimento é difícil e não existe um padrão de atribuição de valores do limiar que torne fácil a tarefa de excluir os pontos da sombra na segmentação. A Figura 8 mostra o tipo de resultados obtidos para implementação destes algoritmos. Existem situações para as quais é possível isolar as sombras, entretanto pequenas mudanças na iluminação tornam os limiares escolhidos inadequados, fazendo com que estes algoritmos não tenham um desempenho aceitável.

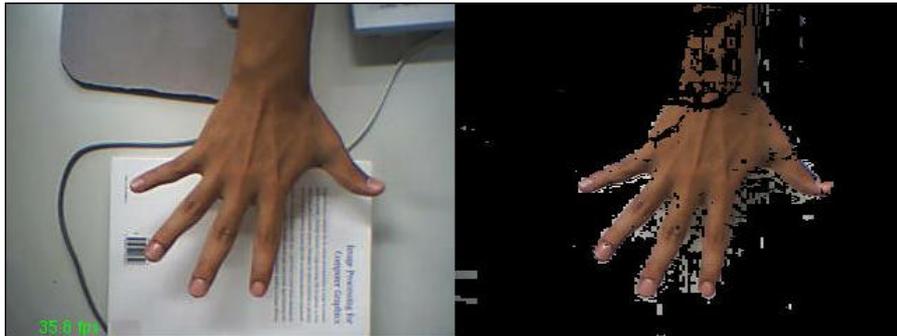


Figura 8 - Resultado dos algoritmos que lidam parcialmente com sombras; dificuldade na atribuição do limiar e impossibilidade de isolar as sombras; morfologia deformada do objeto.

Como podemos notar na Figura 8, existem alguns pixels em que há diminuição da iluminação devido à oclusão da luz pela mão; estes pixels em sombra não mudam de cor, mas experimentam uma redução de luminosidade. Portanto é indispensável considerar modelos que permitam saber que houve apenas mudança na iluminação e não na cor original do fragmento da imagem. Dentro do estudo sobre algoritmos de subtração de fundo, encontramos dois algoritmos que conseguem fazer esta distinção: o algoritmo de Horprasert et al. [14, 15], que calcula a distorção do brilho e a distorção de cor de cada ponto no espaço RGB, e o algoritmo de Cheung et al. [3], que calcula o ângulo formado entre os dois vetores que representam pontos no espaço RGB.

Cada um destes algoritmos teve um desempenho aceitável para os requisitos do presente estudo. Na seção seguinte apresentamos estes algoritmos em mais detalhes procurando justificar a escolha do algoritmo de Horprasert et al. [14].

3.3. Algoritmos Robustos de Subtração de Fundo

Antes de discutirmos os algoritmos é importante fazermos algumas considerações para entendermos algumas características físicas que influenciam as câmeras de vídeo. Os sensores utilizados nas câmeras tipicamente transformam cores de um espaço de dimensão infinita em cores em um espaço tridimensional RGB através de filtros de cores nas bandas vermelha, verde e azul. Além da perda de precisão na digitalização das cores, existe uma contínua variação de cor ao longo do tempo. Raramente é possível obter o mesmo valor RGB de cor num ponto específico da imagem durante um certo período de tempo. Ruídos inerentes aos equipamentos e as flutuações de iluminação

presentes nas fontes de luz fazem variar estes valores. Além disso, existe uma variação desigual nos filtros de cores nas bandas vermelha, verde e azul, o que indica que existe uma sensibilidade diferente para cada uma destas bandas (a variação é diferente em cada canal de cor).

Uma das habilidades fundamentais no sistema de visão humano é a constância de cor. Os humanos tendem a atribuir uma cor constante a um objeto, embora existam mudanças na iluminação ao longo do tempo ou espaço. O algoritmo de Horprasert et al. [14, 15] e o algoritmo de Cheung et al. [3] tentam reproduzir esta habilidade na modelagem e subtração do fundo na cena.

O esquema básico de subtração de fundo nestes algoritmos consiste em subtrair a imagem atual da imagem de referência, a qual modela o fundo. Para que isto possa ser feito, os algoritmos começam com uma etapa de modelagem na qual se constrói a imagem de referência; em seguida, há uma etapa estatística para determinar um limiar apropriado nas variáveis de comparação e finalmente a etapa de subtração e classificação, em que cada ponto é segmentado em grupos diferentes. A segunda etapa não está explicitamente indicada no algoritmo de Cheung et al. [3]; portanto, a seleção do limiar é feita unicamente por um processo de tentativa e erro (testando diferentes valores).

Antes de apresentarmos cada uma das etapas dos algoritmos é importante mostrarmos o modelo computacional de cor que eles utilizam. Ambos algoritmos citados acima trabalham com valores de cores no espaço tridimensional RGB.

3.3.1. Modelo de Cor

No algoritmo apresentado por Horprasert et al. [14], o modelo é baseado na diferença entre as cores medida através de distorções de brilho (*brightness*) e de cromaticidade (*chromaticity*). A Figura 9 ilustra o modelo computacional de cor proposto neste algoritmo no espaço tridimensional RGB. Considerando o ponto (*pixel*) i na imagem, denota-se $E_i = [E_R(i), E_G(i), E_B(i)]$ como sendo o vetor que representa a cor esperada do ponto (imagem de referência). A linha $\overline{OE_i}$, que passa pela origem e através do ponto E_i , é chamada de *linha de cor* (*expected chromaticity line*). $I_i = [I_R(i), I_G(i), I_B(i)]$ é a cor RGB do ponto na imagem atual, da qual se quer subtrair o fundo. Neste modelo basicamente se quer medir a distorção existente entre I_i e E_i . Isto é possível medindo a distorção de brilho e de cor, no valor RGB do pixel.

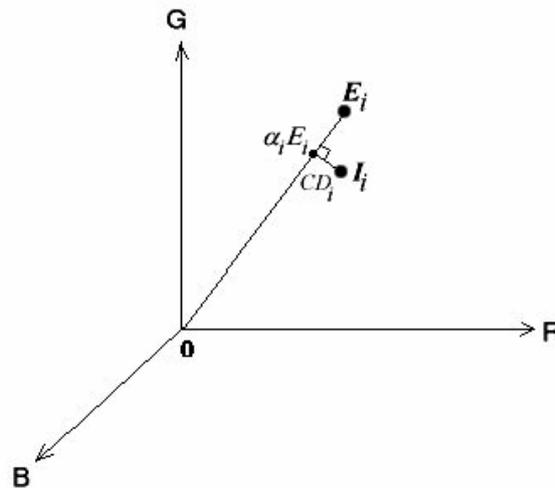


Figura 9 - Modelo de cor no espaço tridimensional RGB proposto em [14]. E_i representa o valor de cor esperado do ponto i na imagem de referência, I_i representa a cor na imagem atual. A diferença entre I_i e E_i é decomposta em distorção de brilho (a_i) e distorção de cor (CD_i). (Extraída de [14]).

A distorção de brilho (a_i) é o valor escalar que projeta a cor atual I_i na linha de cor de modo que ela se aproxime o máximo possível do valor de referência E_i . a_i é o argumento que minimiza:

$$f(a_i) = (I_i - a_i E_i)^2$$

O valor a_i representa a distorção de brilho de I_i com relação ao valor esperado E_i ; se a_i é menor do que 1 este é mais escuro, se a_i é maior do que 1 este é mais claro. A distorção de cor CD_i é definida como sendo a distância ortogonal entre o valor atual I_i e a linha de cor no modelo. Este é dado por:

$$CD_i = \|I_i - a_i E_i\|$$

Para entender melhor este modelo, são importantes algumas considerações. A distorção de brilho é um valor escalar que posiciona o valor da cor atual ao longo da linha de cor. Uma característica importante desta linha é que ela mantém um certo padrão de cor ao longo dela, mudando apenas a intensidade luminosa. Assim é possível determinar se um certo ponto na imagem tem um valor de cor parecido com outro, mas com brilho diferente. A distorção de brilho significa apenas uma mudança na iluminação e não uma mudança

significativa na cor do pixel. Esta diferenciação permite identificar pontos da imagem que representam sombras, pois eles têm um brilho bem menor do que o original no modelo de referência, mas ainda mantêm uma cor parecida à original. A distorção de cor indica o grau de afastamento que se tem dos padrões de cores representados pela linha. Assim, através deste indicador pode-se saber se a cor atual está muito longe ou não do padrão de cores representado pela linha, o que é útil para saber se aquele pixel faz parte de um objeto diferente que está presente na imagem atual.

No algoritmo de subtração de fundo apresentado por Cheung et al. [3] não existe um modelo específico de decomposição de cores. Os valores de cor são considerados como sendo os valores RGB, e as etapas de modelagem e subtração de fundo consideram os valores de cor neste espaço. Portanto, os valores de cor não são decompostos nem transformados neste algoritmo. O fato de que o algoritmo meça o ângulo entre dois vetores no espaço RGB indica que a comparação é feita entre duas linhas de cor.

3.3.2. Modelagem do Fundo

No processo de treinamento do algoritmo apresentado por Horprasert et al. [14], a imagem de referência e alguns atributos relacionados com ela são calculados ao longo de um período de tempo ou com base em um número definido de imagens de treino. Cada pixel é modelado como sendo um vetor de 4 componentes, $\langle E_i, s_i, a_i, b_i \rangle$, onde E_i é o valor esperado da cor, s_i é o desvio padrão, a_i é a distorção média de brilho e b_i é a distorção média de cor. Neste algoritmo, E_i é considerado como sendo a imagem média das imagens de treino e s_i como sendo o desvio médio padrão presente neste conjunto de imagens. Ou seja:

$$E_i = [m_R(i), m_G(i), m_B(i)]$$

$$S_i = [s_R(i), s_G(i), s_B(i)]$$

Para o cálculo de a_i e b_i são consideradas as seguintes equações [14]:

$$\mathbf{a}_i = (\arg \min) \left[\left(\frac{I_R(i) - \mathbf{a}_i \mathbf{m}_R(i)}{\mathbf{s}_R(i)} \right)^2 + \left(\frac{I_G(i) - \mathbf{a}_i \mathbf{m}_G(i)}{\mathbf{s}_G(i)} \right)^2 + \left(\frac{I_B(i) - \mathbf{a}_i \mathbf{m}_B(i)}{\mathbf{s}_B(i)} \right)^2 \right] \quad (1)$$

$$= \frac{\left(\frac{I_R(i) \mathbf{m}_R(i)}{\mathbf{s}_R(i)^2} + \frac{I_G(i) \mathbf{m}_G(i)}{\mathbf{s}_G(i)^2} + \frac{I_B(i) \mathbf{m}_B(i)}{\mathbf{s}_B(i)^2} \right)}{\left(\left[\frac{\mathbf{m}_R(i)}{\mathbf{s}_R(i)} \right]^2 + \left[\frac{\mathbf{m}_G(i)}{\mathbf{s}_G(i)} \right]^2 + \left[\frac{\mathbf{m}_B(i)}{\mathbf{s}_B(i)} \right]^2 \right)}$$

$$CD_i = \sqrt{\left(\left(\frac{I_R(i) - \mathbf{a}_i \mathbf{m}_R(i)}{\mathbf{s}_R(i)} \right)^2 + \left(\frac{I_G(i) - \mathbf{a}_i \mathbf{m}_G(i)}{\mathbf{s}_G(i)} \right)^2 + \left(\frac{I_B(i) - \mathbf{a}_i \mathbf{m}_B(i)}{\mathbf{s}_B(i)} \right)^2 \right)} \quad (2)$$

$$a_i = RMS(\mathbf{a}_i) = \sqrt{\frac{\sum_{i=0}^N (\mathbf{a}_i - 1)^2}{N}} \quad (3)$$

$$b_i = RMS(CD_i) = \sqrt{\frac{\sum_{i=0}^N (CD_i)^2}{N}} \quad (4)$$

O processo de treino e o cálculo das imagens de referência do fundo, no algoritmo de Horprasert et al. [14], é mostrado na Figura 10. Em resumo, o processo começa com a captura de um número de imagens de treino, a partir das quais calcula-se uma imagem média e outra representando o desvio padrão (Passo 1). Para cada uma das imagens de referência calculam-se as imagens da distorção de brilho e da distorção de cor em relação à imagem média (Passos 2 e 3); para obter estas imagens utilizam-se as Equações (1) e (2) respectivamente. Para o cálculo destas imagens são utilizados a imagem original, a imagem média, o desvio padrão e as Equações (1) e (2), completando-se assim os Passos 2 e 3 indicados na Figura 10. Para o Passo 5 utiliza-se a Equação (3) e para o Passo 4 utiliza-se a Equação (4), completando assim o processo de treinamento.

No final do processo de treino no algoritmo [14] são obtidas 4 imagens de referência: a imagem média, a imagem do desvio padrão médio, a imagem da

distorção média de brilho e a imagem da distorção média de cor. Nestas imagens estão modeladas as características do fundo e são estas as utilizadas como referência no processo de subtração.

Observando-se as Equações (1) e (2), onde se calculam a distorção de brilho e de cor, pode-se ver que, no cálculo, estes valores são normalizados ou divididos pelo valor do desvio padrão [14]; assim, pixels com baixo desvio padrão podem ter valores de distorção de brilho e cor muitos altos. Para solucionar este problema é importante determinar um valor mínimo do desvio padrão que pode ter cada um dos pontos nas imagens. Não existe um padrão de atribuição para este valor no algoritmo [14], portanto este valor é atribuído por um método de tentativa e erro (testando diferentes valores positivos menores ou iguais a 1) como foi feito aqui, na implementação do algoritmo. Os valores utilizados foram os mesmos para cada banda de cor. Estes variaram de 0.7 até 1.0 no espaço de cores de 0 até 255.

No processo de treinamento do algoritmo apresentado em [3] não existe um modelo detalhado de modelagem do fundo. A única imagem de referência utilizada por este algoritmo é a imagem média, esta é calculada a partir de um determinado conjunto de imagens de treino; por tanto, só é preciso o cálculo da imagem média no espaço RGB.

3.3.3. Subtração ou Segmentação

Nesta parte do processo é calculada a diferença existente entre a imagem de referência e a imagem atual. No algoritmo apresentado por Horprasert et al. [14] essa diferença é decomposta em distorção de brilho e distorção de cor. Aplicando um limiar apropriado na comparação da distorção de brilho e da distorção de cor em um pixel, pode-se obter uma máscara indicando qual é o tipo do pixel que está sendo comparado. De acordo com o modelo de decomposição de cor que é utilizado na fase de subtração, cada pixel pode ser classificado em uma das 4 categorias seguintes:

- **Fundo original (B)**, se a distorção de brilho e a distorção de cor são similares aos valores correspondentes na imagem de referência.
- **Fundo com baixa iluminação ou sombra (S)**, se a distorção de cor é similar, mas a distorção de brilho é bem menor do que aquela na imagem de referência.

- **Fundo como alta iluminação (H)**, se existe uma distorção de cor similar, mas uma distorção de brilho maior do que aquela na imagem de referência.
- **Objeto em movimento (F)**, se a distorção de cor é diferente aos valores esperados no treinamento, caso em que o pixel é considerado como sendo parte do objeto em movimento.

A Figura 10 ilustra a etapa de subtração deste algoritmo. A imagem que contém o objeto é comparada com a imagem média e o desvio padrão para obter imagens que representam a distorção de brilho e a distorção de cor (Passo 6), para o que aplicam-se as Equações (1) e (2) respectivamente. As imagens são comparadas com as imagens de referência aplicando valores apropriados do limiar (*threshold*) de comparação (Passo 7). No caso do algoritmo implementado neste trabalho foram considerados apenas dois grupos: fundo e objeto em movimento. Pontos pertencentes a sombras ou pontos com maior iluminação foram considerados como sendo parte do fundo. A comparação da imagem atual com a imagem de referência é feita basicamente com base nas diferenças existentes entre os valores das distorções de brilho e das distorções de cor.

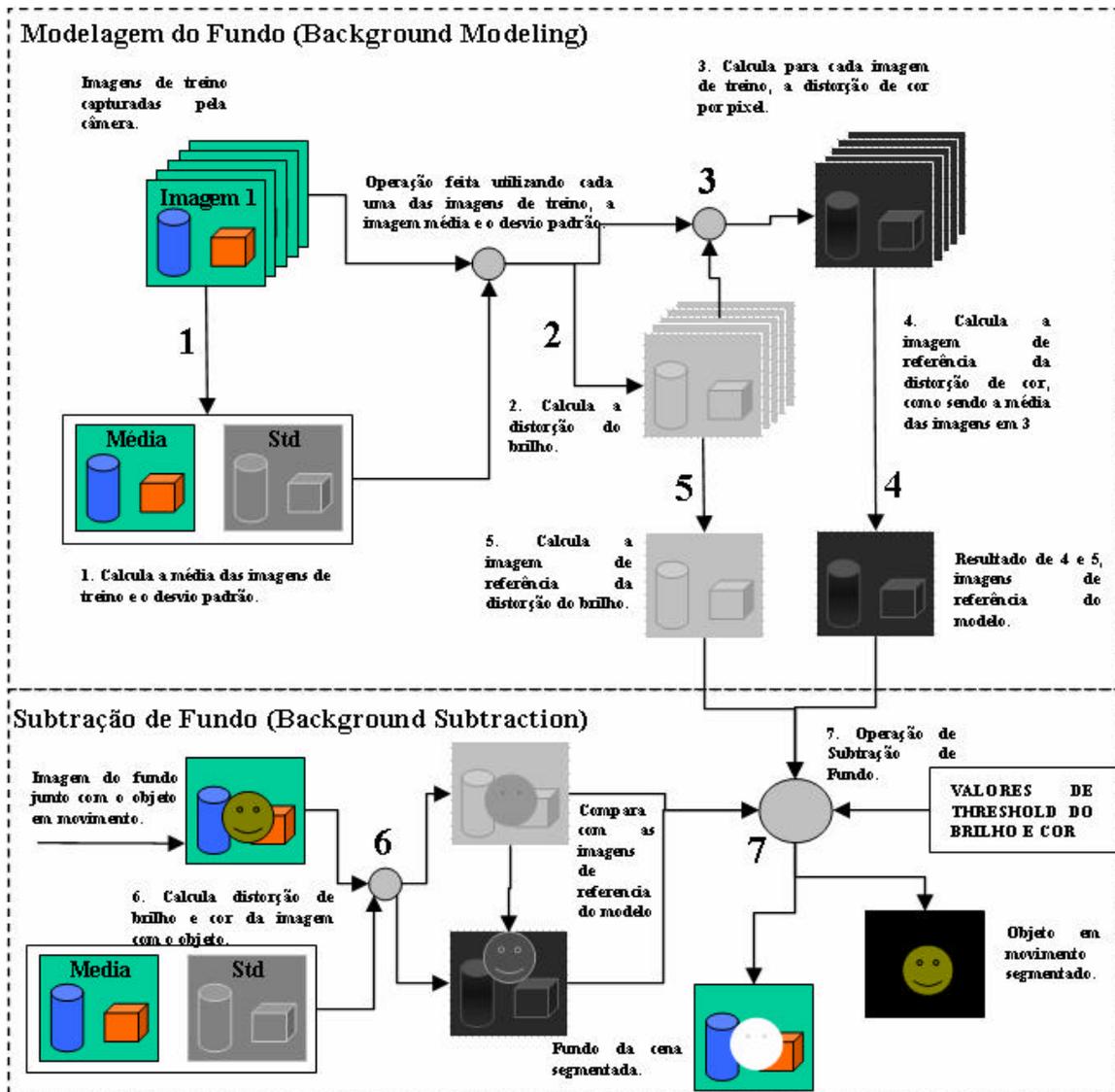


Figura 10 - Algoritmo de subtração de fundo baseado na decomposição da diferença de cor em distorção de brilho e distorção de cor. Seguindo a ordem das operações indicada pelos números, mostram-se as etapas de modelagem e subtração de fundo. A etapa de modelagem é feita uma vez, enquanto que a etapa de subtração é feita a cada imagem (*frame*).

Como foi mencionado anteriormente, o modelo de cor deste algoritmo é muito mais sensível à distorção de brilho do que à distorção de cor. Por esta razão é mais fácil identificar as sombras como sendo pontos que mudam apenas seu brilho nas imagens. Assim como a mudança na distorção do brilho é maior do que a mudança na distorção de cor, existem diferentes pixels na imagem que têm diferentes distribuições de a_i e CD_i . Isto significa que os padrões de mudança não são iguais em todos os pixels. A fim de usar um limiar só para todos os pixels, é preciso normalizar os valores de a_i e CD_i com respeito aos

valores médios da distorção de brilho a_i e a distorção de cor b_i . A distorção de brilho normalizada b_i e a distorção de cor normalizada d_i , respectivamente, são:

$$b_i = \frac{a_i - 1}{a_i}$$

$$d_i = \frac{CD_i}{b_i}$$

Com base nestas definições, segundo o algoritmo de Horprasert et al. [14] podemos classificar o pixel em uma das quatro categorias **B**, **S**, **H**, **F** diferentes, através da seguinte regra de decisão:

$$M(i) = \begin{cases} F : d_i > t_{CD}, \text{ senão} \\ B : b_i < t_{a1} \text{ e } b_i > t_{a2}, \text{ senão} \\ S : b_i < 0, \text{ senão} \\ H : \text{caso contrário} \end{cases}$$

onde t_{CD} , t_{a1} e t_{a2} são os valores do limiar utilizados para determinar as similaridades de brilho e cor existentes entre as imagens de referência do modelo e a imagem atual que está sendo observada.

No entanto, de acordo com o modelo de cor utilizado neste algoritmo, é possível que pixels do objeto em movimento que contenham valores RGB próximos da origem (objetos pretos ou partes pretas dos objetos) sejam classificados como sendo sombras, portanto como sendo parte do fundo. Isto ocorre porque o fato de que todas as linhas de cor no espaço RGB passam através da origem faz com que estas cores sejam consideradas parecidas com qualquer linha de cor. Para evitar este problema foi introduzido um novo limiar [14] para a distorção normalizada do brilho (t_{slo}), que indica o limite de distorção de brilho que deve ter um pixel para ser ou não ser considerado como sombra. Desta forma, a regra de decisão ficaria como:

$$M(i) = \begin{cases} F : d_i > t_{CD} \text{ ou } b_i < t_{alo} \text{ senão} \\ B : b_i < t_{a1} \text{ e } b_i > t_{a2}, \text{ senão} \\ S : b_i < 0, \text{ senão} \\ H : \text{caso contrário} \end{cases}$$

A Figura 11 ilustra o espaço bidimensional definido pela distorção normalizada de brilho b e pela distorção normalizada de cor d . Este espaço é partilhado pelos limiares utilizados na etapa de subtração e representa graficamente a regra de decisão utilizada para classificar cada um dos pixels da imagem. Cada uma das 4 categorias **F**, **B**, **S** e **H** são representadas. No nosso caso as sombras (**S**) e as regiões com alta iluminação (**H**) foram consideradas como sendo o fundo (**B**).

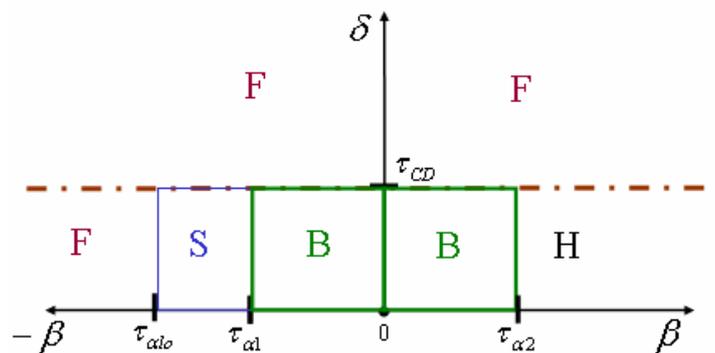


Figura 11 – Regiões definidas pelos limiares no espaço bidimensional definido pela distorção normalizada de brilho b e a distorção normalizada de cor d .

Na etapa de subtração do algoritmo de Cheung et al. [3], a diferença entre as imagens é calculada através do ângulo formado entre cada uma das cores na imagem de referência e da imagem atual. O mecanismo principal de subtração de fundo utilizado neste algoritmo é baseado no cálculo do ângulo, mediante o qual os pontos da imagem são classificados. Na Figura 12, mostram-se os passos do processo de subtração de fundo feito neste algoritmo. A idéia principal é que alguns pontos da imagem sejam excluídos do teste do ângulo através de testes rápidos de variação de cor (distância entre as cores $d(i,j)$). Se existe uma variação grande de cor, acima de um limiar estabelecido T_U , o algoritmo supõe que a mudança neste ponto é consideravelmente alta, e conseqüentemente considera o ponto como sendo parte do objeto (Passo 1). Se não existe uma mudança considerável na cor, então pode ser que os valores de cor nos pontos não tenham sofrido grandes mudanças. Deste modo, se a variação de cor é menor do que um outro limiar T_L o ponto é considerado como sendo parte do fundo (Passo 2). Já para os pontos que apresentam variações de cor acima do T_L e abaixo do T_U o ângulo deve ser medido. Com base no critério de comparação entre o limiar e o ângulo entre os vetores, pode-se classificar os

pontos dentro de duas classes: se $q < T_C$ então o pixel pertence ao fundo, caso contrário pertence ao objeto. No presente algoritmo T_U e T_L representam a variação máxima e mínima de cor para que os pontos sejam considerados como sendo o objeto ou o fundo respectivamente. O valor do limiar do ângulo T_C indica o grau de semelhança que as cores devem ter para que possam ser consideradas como parecidas ou diferentes.

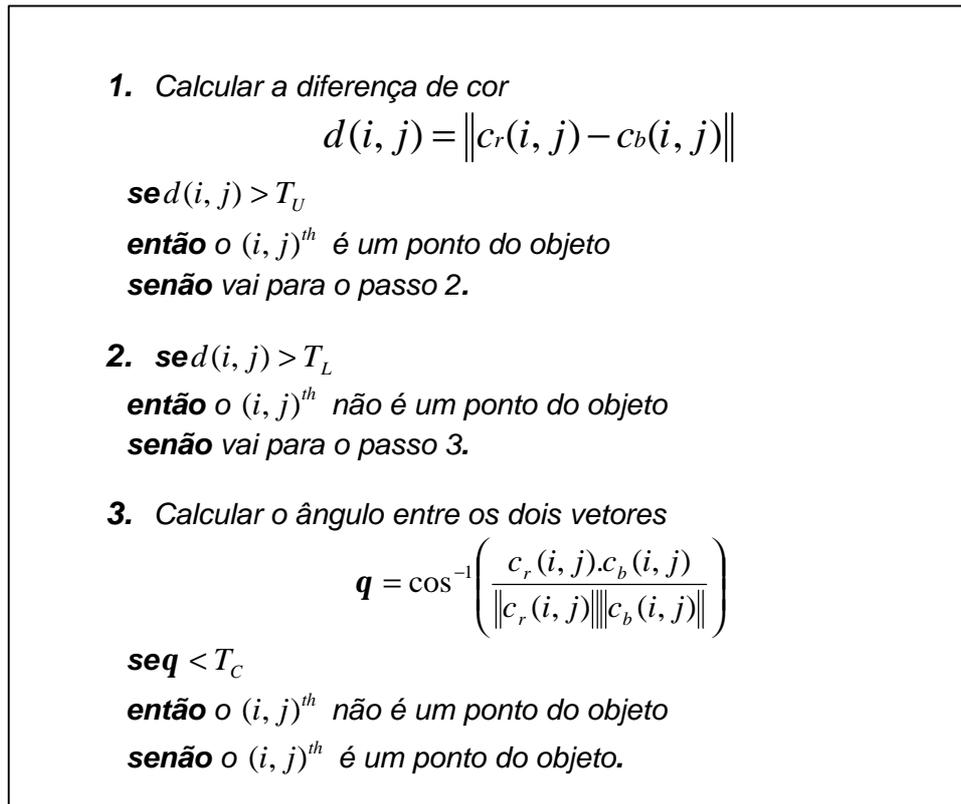


Figura 12 - Algoritmo de geração de silhuetas ou de subtração de fundo, baseado no ângulo entre dois vetores. C_r é a cor atual e C_b é a cor de referência.

3.3.4. Seleção dos Limiares

Na seleção automática dos limiares calculam-se valores dos limiares para conseguir melhores resultados no processo de subtração. Tradicionalmente este procedimento é feito com base em um aprendizado estatístico, com imagens que ainda pertencem à etapa de modelagem do fundo.

No algoritmo apresentado por Cheung et al. [3] não existe um procedimento específico para atribuir valores ao limiar do ângulo. Pelo fato de ser um único limiar, não é muito complicado conseguir um valor adequado através da seleção de valores, até que se alcancem resultados razoáveis.

Já no algoritmo apresentado por Horprasert et al. [14], existe a necessidade de se encontrarem valores para o limiar, tanto para a distorção normalizada de brilho quanto para a distorção normalizada de cor. Pelo fato de serem dois os limiares que devem ser determinados, um procedimento de ajuste do limiar poderia resultar em um trabalho difícil de se realizar, pois existe um conjunto grande de combinações destes valores que podem ser atribuídos. Num processo de seleção de limiares tenta-se diminuir este conjunto (espaço) de combinações de valores ao mínimo possível ou até que se encontrem diretamente os valores que devem ser atribuídos a estes limiares. A grande vantagem do processo de seleção de limiares é restringir o espaço de busca dos limiares para valores próximos dos limiares que finalmente serão utilizados. Encontrar sub-regiões do espaço de busca, onde os valores ocasionem um melhor processo de subtração, é uma das finalidades do processo de seleção de limiares. Sem uma estratégia de busca muitas vezes acontece de se testar valores de limiar sem saber realmente se estes valores, como um par de limiares, são discriminantes o suficiente para serem utilizados.

No processo de seleção dos limiares no algoritmo proposto em [14], os limiares para a distorção normalizada do brilho e para a distorção normalizada de cor são calculados através de um processo de aprendizado estatístico. Este aprendizado consiste em montar um histograma para a distorção normalizada do brilho e para a distorção normalizada de cor. Para montar este histograma são calculadas as distorções normalizadas do brilho e de cor de todas as imagens de treino em relação à imagem média. Tendo estes valores, pode-se montar o histograma que representa o padrão de mudança das distorções ao longo do processo de treinamento ou modelagem do fundo; em outras palavras, qualquer imagem que tenha objetos diferentes aos modelados no fundo vai ter um histograma relativamente diferente. Na Figura 13 mostra-se a curva típica dos histogramas da distorção normalizada de brilho e a distorção normalizada de cor respectivamente.

Olhando para os histogramas pode-se ver um determinado intervalo onde flutuam os valores das distorções. Assim, se existirem valores fora deste intervalo pode-se supor que esses pixels fazem parte de um objeto que não estava na modelagem do fundo. Uma vez que os histogramas são construídos, os valores dos limiares são automaticamente selecionados de acordo com a taxa de detecção (valor em porcentagem) que se quer obter. Por exemplo 100% significa que todas as amostras pertencem ao intervalo. O limiar para a distorção normalizada de cor, t_{CD} , é o valor da distorção normalizada de cor na taxa de

detecção r desejada (ex. 4.06 corresponde a 99.95%). Na distorção normalizada de brilho existem dois valores (t_{a1} e t_{a2}) e por isto deve-se definir um intervalo no histograma. t_{a1} é o valor na taxa r desejada e t_{a2} é o valor na taxa $(100\% - r)$.

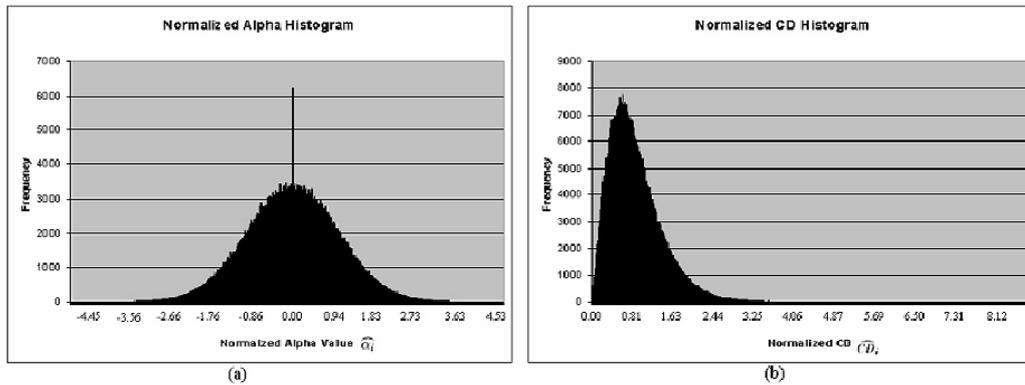


Figura 13 - (a) histograma da distorção normalizada de brilho (\hat{b}), e (b) histograma da distorção normalizada de cor (\hat{d}). Extraído de [14].

Neste processo de seleção de limiares existem algumas considerações importantes que devem ser feitas. Como foi afirmado anteriormente, o valor da distorção de brilho e da distorção de cor é normalizado pelo valor do desvio padrão. Isto traz alguns problemas na hora de calcular as distorções, pois podem ter valores muito elevados. Já que o valor do desvio padrão e a natureza das variações deste dependem muito dos equipamentos que são utilizados na captura, os valores calculados das distorções também devem incluir estas características. Por isto, nas implementações realizadas deste algoritmo, os valores do limiar dependem dos equipamentos utilizados e até das condições particulares em que foram testados. Sugere-se deste modo construir os histogramas para o cálculo dos limiares e desconsiderar os valores de limiares sugeridos em outros trabalhos.

No cálculo e construção dos histogramas são utilizados os valores das distorções de brilho e de cor normalizados. Na etapa de subtração os valores comparados também são os valores normalizados das distorções. Para os pixels na imagem que têm distorções de cor e brilho muito baixos, existe um problema na hora de normalizar: o valor da distorção normalizada pode ser muito elevado e assim qualquer valor do limiar é ultrapassado, produzindo falsos pontos do objeto na etapa de subtração. Uma solução para este problema é atribuir valores de mudança mínima tanto para a distorção de brilho quanto para a distorção de

cor, como foi feito no desvio padrão mínimo dos pixels. Todos estes problemas devem ser levados em conta tanto ao se construir e calcular os histogramas das distorções quanto no cálculo das distorções normalizadas na etapa de subtração. Se estes problemas não forem considerados é difícil obter curvas parecidas às da Figura 13. A construção de curvas relativamente parecidas às da Figura 13, sem variações bruscas e com tendências suaves, sugere um indicador muito útil para a avaliação dos valores mínimos atribuídos às distorções e ao desvio padrão.

3.4. Resultados

Nesta seção são apresentados os resultados obtidos da implementação dos algoritmos apresentados por Horprasert et al. [14] e Cheung et al. [3]. Os testes foram realizados em seqüências de imagens de vídeo com resolução de 320x240 pixels, capturadas por uma câmera WebCam Logitech QuickCamPro 4000 conectada via interface USB em um computador com processador Pentium III de 800Mhz e com 128 MB de memória RAM. Os modelos que lidam com as sombras e flutuações na iluminação permitem resultados melhores que os mostrados na Figura 8. Isto ocorre porque são mais sensíveis a mudanças no brilho do que as mudanças na cor, conseguindo assim um resultado muito próximo da morfologia do objeto "MÃO".

Nas imagens que seguem são mostrados o tipo de resultados alcançados na implementação de cada um dos dois algoritmos. Estas imagens mostram um frame determinado da seqüência, mas representam o tipo e a qualidade de resultados obtidos ao longo da seqüência de vídeo testada. Além disto, a disposição do plano de fundo com relação à mão permite mostrar as sombras geradas, as quais, servem para testar o desempenho dos algoritmos frente à presença marcante das sombras dos objetos. Nos resultados não foi feito nenhum tipo de pós-processamento para melhorar a qualidade da segmentação. As imagens representam unicamente o tipo de resultado obtido na etapa de subtração dos algoritmos testados. Pós-processamentos são temas para etapas posteriores.

Na Figura 14 são apresentados os resultados obtidos pelo algoritmo de subtração baseado na medida do ângulo entre as cores [3]. A imagem da cena sem o objeto de interesse não é mostrada, mas fica claro na figura qual é a imagem do fundo. O objeto é segmentado com as suas cores originais dando

uma impressão diferente da imagem da segmentação binária, onde a morfologia do objeto é mais fácil de perceber. Para fins práticos a imagem de segmentação binária é muito mais importante, e a qualidade desta influi muito nos processamentos posteriores que serão feitos. Nos testes efetuados, o limiar atribuído ao ângulo foi $q = 17.49^\circ$. As sombras não são totalmente excluídas do objeto e deformam a sua morfologia, o qual não é desejável. Dependendo do limiar atribuído ao ângulo, as sombras somem, porém partes que realmente pertencem ao objeto também somem. Por isto é difícil determinar um limiar que após um determinado valor não cause a deformação do objeto.

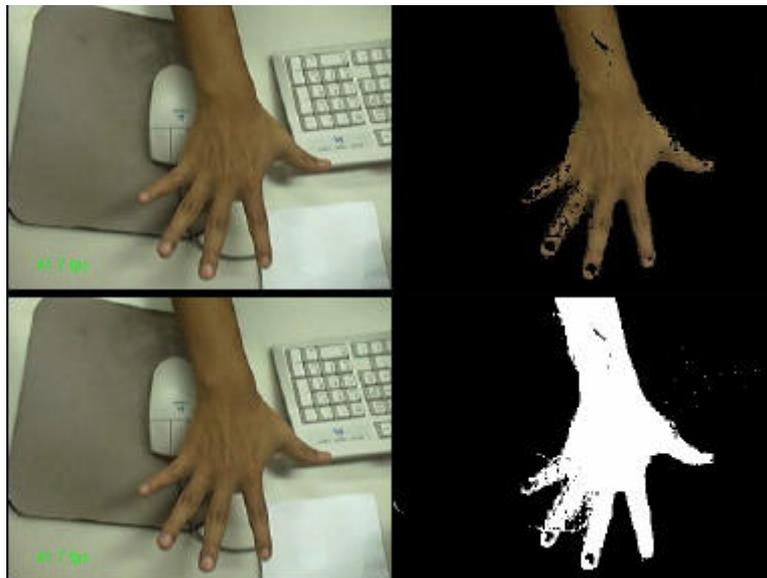


Figura 14 - Resultado da segmentação do algoritmo baseado na medida do ângulo [3]. O limiar utilizado para o ângulo foi de $q = 17.49^\circ$. Nas figuras à esquerda são mostradas cenas com a mão e nas figuras à direita são mostrados os resultados da segmentação. Nos resultados são apresentados a segmentação com as cores originais do objeto e a segmentação binária, onde é mais fácil perceber a morfologia do objeto segmentado. O número de quadros por segundo alcançado foi igual a 30.

Na Figura 15 são mostrados os resultados obtidos pelo algoritmo de subtração de fundo baseado na medida da distorção de brilho e a distorção de cor entre duas cores [14]. Como o objetivo é segmentar o objeto, o interesse está focado unicamente no objeto e não em detectar sombras ou partes com mudanças na iluminação; para isto só serão utilizados os parâmetros de limiar t_{CD} para a distorção de cor e t_{lo} para a distorção de brilho. Aos parâmetros de desvio padrão mínimo (s'), distorção mínima de cor (CD') e distorção mínima de brilho (a') foram atribuídos os valores $s' = 1.0$, $CD' = 0.75$ e $a' = 0.01$

respectivamente. Estes valores foram mantidos tanto para as etapas de modelagem do fundo e construção dos histogramas quanto para a etapa de subtração de fundo. Os únicos limiares necessários para segmentar o objeto são t_{CD} para a distorção de cor e t_{LO} para a distorção de brilho. Os valores atribuídos para estes limiares foram $t_{CD}=14.1$ e $t_{LO}=-80.5$. Claramente se pode observar que este algoritmo possui uma segmentação melhor do que o algoritmo baseado na medida do ângulo entre dois vetores [3] (Veja-se a Figura 14); o fato de ter dois parâmetros na comparação ajuda muito na eficácia deste algoritmo. Por este motivo, o algoritmo baseado na medida da distorção de brilho e na distorção de cor [14] foi escolhido como o método de segmentação a ser utilizado neste trabalho.



Figura 15 - Resultados do algoritmo baseado na medida da distorção de brilho e a distorção de cor [14]. Desvio padrão mínimo ($s'=1.0$), distorção mínima de cor ($CD'=0.75$), distorção mínima do brilho ($a'=0.01$). Limiares de comparação utilizados na etapa de subtração: $t_{CD}=14.1$ para a distorção de cor, $t_{lo}=-80.5$ para a distorção de brilho. Número de quadros por segundo igual a 43.

Com o objetivo de melhorar o desempenho do algoritmo baseado na distorção de brilho e de cor [14], foi efetuada uma pequena modificação para reduzir o número de pixels testados em cada iteração do processo de segmentação. Ao invés de serem testados todos os pixels da imagem, são testados apenas alguns pixels representativos para cada janela de 2x2 pixels. Se este ponto é considerado pelo algoritmo como sendo parte do objeto, então os demais pontos na janela também são considerados como sendo parte do objeto;

caso contrário os 4 pontos são considerados como sendo parte do fundo. Este teste simples não piora muito a segmentação do objeto (Veja-se Figura 16), mas tem um ganho muito grande no número de quadros por segundo, alcançando a casa dos 60 qps.



Figura 16 - Teste do algoritmo em [14] aplicando uma janela de 4 pixels. Esta mudança aumenta consideravelmente o número de quadros por segundo sem alterar significativamente a segmentação do objeto. 60 qps.

3.5. Conclusões

O processo de segmentação de objetos em movimento através de algoritmos de subtração de fundo é bastante complexo. Um dos grandes problemas é que existem muitos parâmetros que afetam sua eficiência e robustez. Conseguir um balanceamento ótimo destes parâmetros é um problema difícil e só é possível conseguir uma aproximação dos resultados ótimos de segmentação. Entendam-se por parâmetros os valores e limiares que devem ser obtidos pelo algoritmo. As combinações e o espaço de valores que eles podem assumir é uma das principais causas da dificuldade de se determinar um balanceamento ótimo entre eles.

Outro problema importante é que os algoritmos de subtração de fundo dependem muito das condições da iluminação, formação de sombras e flutuações das fontes de luz, com as quais nem todos os algoritmos estudados puderam lidar satisfatoriamente. Mesmo sob condições controladas, é muito difícil conseguir uma segmentação próxima a do objeto original.

Outra questão que teve que ser considerada é a capacidade de processamento em tempo real, que limita muitas vezes a inclusão de modelos mais sofisticados e robustos. É possível, entretanto, que tais métodos possam

ser utilizados em sistemas de processamento paralelo, o que não é o objetivo deste trabalho.

Dentro deste estudo só foram considerados algoritmos que trabalham com a modelagem do fundo estático de uma cena.

Apesar de todas as dificuldades, os estudos e os testes realizados demonstram que no presente trabalho foram conseguidos resultados satisfatórios em todos os requisitos necessários para a continuação do desenvolvimento do sistema de reconhecimento que é tema desta dissertação.