

Implementação de um Extrator de Características baseado em Momentos da Imagem

Kleber Padovani Souza e Hemerson Pistori
Grupo de Pesquisa em Engenharia e Computação, GPEC
Universidade Católica Dom Bosco, UCDB
Av. Tamandaré, 6000 - Jardim Seminário
79117-900 Campo Grande, MS
(kriowloo, pistori)@ec.ucdb.br

Abstract

Image moments and its functions have been used in many computer vision systems during the feature extraction phase, mainly as a mean to achieve scale, translation and rotational invariance. This paper describes the implementation of a general-use feature extractor based on image moments. This implementation, besides being free and compatible with the ImageJ package, has some graphical resources that make the concepts related to image moments easier to understand. This visual feedback turns this implementation into a good tool for learning and investigation on image feature extraction.

Resumo

Momentos da imagem e funções definidas sobre esses momentos têm sido utilizados em muitas aplicações durante a fase de extração de características, a fim de se alcançar um reconhecimento de objetos que seja invariante à escala, translação e rotação. Este artigo descreve uma implementação que, além de ser livre e compatível com o pacote ImageJ, possui alguns recursos gráficos que facilitam a compreensão de conceitos relacionados com momentos da imagem. A retro-alimentação visual, fornecida por essa implementação, a torna uma boa ferramenta para aprendizagem e investigação na área de extração de atributos em imagens digitais.

1 Introdução

A visão computacional, ou visão de máquina, é uma área de conhecimento que se dedica a desenvolver teorias e métodos voltados à extração automática de informações “úteis” contidas em imagens. A utilidade de uma

informação é altamente dependente da aplicação. Por exemplo, em uma indústria de manufatura, que fabrica determinado produto, uma informação útil pode estar relacionada com alguma diferença de cor ou forma padrão, esperada para um produto sem defeitos. Tarefas relativamente simples de serem realizadas por seres humanos, como diferenciar chaves de fenda de chaves alemãs, em uma linha de montagem, apresentam grandes desafios para sistemas automáticos de visão computacional.

Basicamente, sistemas de visão computacional capturam imagens através de dispositivos optico-eletrônicos, como webcams e filmadoras digitais, e buscam produzir descrições úteis das informações contidas nas imagens. Essas descrições podem ser utilizadas, por exemplo, na classificação de objetos ou no controle automático de algum dispositivo atuador, como um braço robótico ou uma rede de esteira rolantes em uma linha de montagem.

A área de visão computacional também pode contribuir para a inclusão digital de pessoas que possuem necessidades especiais, como deficiências auditivas, por exemplo. Sistemas com interfaces guiadas por sinais visuais servem como formas de realizar essa inclusão. Um editor de textos para pessoas com dificuldades de fala, onde a comunicação gestual do usuário é interpretada pelo computador e é gerada uma saída processada (como a tradução do gesto em linguagem de sinais para linguagem natural), facilitaria a utilização dos computadores por este tipo de usuário.

Com o intuito de realizar a inclusão digital dos portadores de necessidades especiais, tais como os tetraplégicos e os surdos, que não podem ou possuem certas dificuldades em utilizar os computadores através dos meios comuns de interação homem-máquina, foi criado o projeto SIGUS¹ [8]. Esse projeto surgiu da realidade atual de pessoas com necessidades especiais, que devido às suas dificuldades

¹Informações sobre o projeto SIGUS podem ser obtidas em <http://www.gpec.ucdb.br/sigus>

acabam sendo excluídas do mundo digital, o que tem se tornado cada vez mais necessário no cotidiano das pessoas. O objetivo do SIGUS é o desenvolvimento de um ambiente computacional, de códigos-fonte abertos, que auxilie o desenvolvimento de aplicações que sejam guiadas por interfaces não-convencionais, mais especificamente, guiadas por sinais visuais, como movimentos da mão e da face.

Assim como em diversas aplicações de reconhecimento de padrões, no ambiente SIGUS a etapa de extração de características é um estágio fundamental para caracterização dos objetos contidos nas imagens digitais. Este artigo apresenta um módulo que calcula os momentos da imagem (*Image Moments*) e possibilita a visualização dos resultados obtidos para diferentes imagens. Esse módulo auxiliará o processo de percepção da variabilidade das características entre diferentes objetos, ajudando a determinar quais delas são relevantes para a aplicação. Além desta implementação contribuir com a etapa de seleção de características, ela também pode ser utilizada como uma ferramenta educacional nas disciplinas relacionadas com a área.

A seção 2 deste artigo trata, de maneira breve e geral, da área de reconhecimento de padrões e sua relação com visão computacional. Na seção 3 são apresentados os conceitos, características e aplicabilidade dos momentos da imagem. A seção 4 enfoca a implementação do extrator de características da imagem, descrevendo suas particularidades, tais como suas formas de cálculo diretamente relacionadas com os momentos da imagem e a demonstração dos resultados obtidos. Na seção 5 é exibida a forma de utilização do aplicativo, juntamente com alguns exemplos de imagens utilizadas nos testes e na seção 6 são apresentados a conclusão e os trabalhos futuros.

2 Reconhecimento de Padrões

Reconhecimento de padrões é a disciplina científica cujo objetivo é a criação de teorias e técnicas que permitam a classificação de objetos, ou padrões, dentre um conjunto de categorias ou classes [10]. Dependendo da aplicação, esses objetos podem ser imagens, seqüências de caracteres, sons ou qualquer outro tipo de sinal, geralmente digitalizado, capturado através dos sensores de um sistema computacional. Como em diversas áreas, o advento dos computadores fez crescer a demanda por aplicações de reconhecimento de padrão, que por sua vez gerou novas demandas por desenvolvimentos teóricos mais avançados [10]. A automação na produção industrial e a necessidade por informações manipuladas e recuperadas estão se tornando cada vez mais importantes. Esta tendência tem elevado o reconhecimento de padrões para o topo das aplicações de engenharia e pesquisa dos dias de hoje [10].

O reconhecimento de caracteres, como letras ou números, é um bom exemplo em que as técnicas de reconhe-

cimento de padrões têm sido utilizadas com sucesso. Sistemas de reconhecimento de caracteres óticos (OCR) já são comercializados e utilizados atualmente. Reconhecimentos de padrões em seqüências de DNA ou mesmo em textos não estruturados também são áreas intensamente pesquisadas na atualidade.

Embora o reconhecimento de padrões possa ser aplicado a problemas sem qualquer relação com imagens e visão, existe uma rica intersecção entre essa área e a área da visão computacional, uma vez que o reconhecimento de padrões é uma importante etapa em boa parte dos problemas de visão computacional. Existem algumas etapas que são comumente encontradas em sistemas de visão computacional envolvendo reconhecimento de padrões. Dentre elas encontram-se a segmentação, a extração de características e a classificação. De forma sucinta, segmentação é um processo utilizado para suprimir da imagem o que não tem importância, mantendo-se apenas os objetos de interesse. Após o estágio de segmentação, a imagem processada é enviada para um módulo que calcula ou extrai desses objetos valores que possam servir para discriminá-los, ou seja, características que sirvam para identificar o objeto (como tamanhos, curvaturas, comprimentos, ocorrência de determinadas formas geométricas, etc). Este processo é denominado extração de características e será tratado com maior ênfase nas próximas seções. Por fim, a classificação recebe os valores do módulo extrator de características (muitas vezes representados como vetores de características) e busca enquadrar, com base nesses valores, os objetos de interesse, em uma das classes pré-definidas (e.g. chave de fenda e chave alemã).

3 Momentos da Imagem

Um dos métodos utilizados para a extração das características da imagem são os momentos da imagem, ou momentos estatísticos. Esses momentos e suas funções podem ser caracterizados como valores numéricos, calculados a partir de uma imagem previamente segmentada e são utilizados para fornecer descrições resumidas da distribuição espacial dos pontos contidos na imagem 2-D estudada [1]. Para compreender os momentos da imagem é interessante representar os pixels da imagem, pertencentes ao objeto de interesse, como um conjunto de pontos em um plano cartesiano. Dessa forma, cada pixel de interesse é representado por duas coordenadas, x e y , que descrevem sua posição na imagem, e uma função f que fornece a intensidade (tom de cinza) da imagem no ponto x, y . Em uma imagem binarizada, muito comumente utilizada na extração de atributos por momentos da imagem, a função f possui valor *um* para todos os pixels que pertencem ao objeto de interesse, e valor *zero* para os pixels de fundo (ou que não pertencem ao objeto de interesse).

Os momentos da imagem mais utilizados são os momentos regulares [2], que são definidos pela fórmula:

$$M_{pq} = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} x^p y^q f(x, y) dx dy \quad (1)$$

Nessa fórmula, M_{pq} é o momento de ordem $(p+q)$ da função de intensidade, $f(x, y)$, de uma imagem contínua. Quando estamos utilizando imagens digitais, as integrais dessa fórmula são substituídas por somatórios, com isso, M_{pq} passa para a seguinte forma, em que n_x e n_y representam respectivamente a largura e a altura da imagem digital [6]:

$$m_{pq} = \sum_1^{n_x} \sum_1^{n_y} x^p y^q f(x, y) \quad (2)$$

Além dos momentos regulares, existem também os momentos centrais, que são definidos para imagens digitais pela fórmula:

$$u_{pq} = \sum_1^{n_x} \sum_1^{n_y} (x - x_c)^p (y - y_c)^q f(x, y) \quad (3)$$

Por fim, existem os momentos centrais normalizados, denotados n_{pq} definidos por:

$$n_{pq} = \frac{u_{pq}}{m_{00}^{\frac{(p+q)+1}{2}}} \quad (4)$$

4 Funções Definidas Sobre Momentos da Imagem

A partir dos momentos da imagem definidos na seção anterior, é possível se obter uma grande quantidade de medidas relacionada ao objeto de interesse, e que podem ser úteis na discriminação de diferentes objetos. Os momentos regulares de ordem 0 e 1, por exemplo, podem ser utilizados para calcular o centróide² da imagem, através das seguintes fórmulas:

$$x_c = \frac{m_{10}}{m_{00}}; y_c = \frac{m_{01}}{m_{00}} \quad (5)$$

em que x_c e y_c são as coordenadas do centróide. É interessante notar que m_{00} representa a massa (ou a área do objeto de interesse), enquanto m_{10} e m_{01} representam as projeções dos pontos de interesse nos eixos x e y, respectivamente [2].

Um outro aspecto importante no estudo descritivo de um conjunto de dados é a variabilidade ou dispersão desses dados, relativamente à medida de localização do centro da população. Define-se a variância como sendo a

²O termo centróide também é conhecido como centro de gravidade (ou centro de massa), mas não são sinônimos. O centróide é uma especialização do centro de gravidade para imagens binárias.

medida que se obtém somando-se os quadrados dos desvios das observações da população (pixels), relativamente à sua média, e dividindo-se pelo número de observações da população:

$$\sigma_x^2 = \frac{\sum_1^n (x - x_c)^2}{n} \quad (6)$$

$$\sigma_y^2 = \frac{\sum_1^n (y - y_c)^2}{n}$$

Na fórmula acima, n representa o número total de pixels pertencentes ao objeto de interesse e x_c e y_c são os centros nos eixos x e y, respectivamente. Uma vez que a variância envolve a soma de quadrados, a unidade em que se exprime não é a mesma que a dos dados. Assim, para obter uma medida da variabilidade ou dispersão com as mesmas unidades que os dados, tomamos a raiz quadrada da variância e obtemos o desvio padrão: o desvio padrão é uma medida que só pode assumir valores não negativos e quanto maior for, maior será a dispersão dos dados. Desta forma podemos obter os desvios padrão em x e y da imagem. O cálculo das variâncias nos eixos x e y através de momentos é realizado através das divisões do momento central u_{20} pelo momento regular m_{00} , e do momento central u_{02} pelo momento regular m_{00} , nesta ordem. As raízes quadradas das variâncias em X e Y são os respectivos desvios padrão.

$$\sigma_x^2 = \frac{u_{20}}{m_{00}} \quad (7)$$

$$\sigma_y^2 = \frac{u_{02}}{m_{00}}$$

Uma outra informação interessante que pode ser calculada através de momentos da imagem é a excentricidade de um objeto. A excentricidade é uma característica invariante da imagem, pois é o quociente da divisão entre o maior e o menor eixo da elipse que melhor se ajusta aos pixels da imagem segmentada. Essa elipse pode ser encontrada utilizando-se os momentos centrais de ordem 2 [11]. Através das fórmulas seguintes podemos obter os valores da excentricidade (ϵ) e da orientação (θ) dessa elipse:

$$\tan \theta = \frac{(u_{02} - u_{20} - 2u_{11} + \lambda)}{(u_{02} - u_{20} + 2u_{11} - \lambda)}$$

$$\epsilon^2 = \frac{(u_{20} + u_{02} + \lambda)}{(u_{20} + u_{02} - \lambda)} \quad (8)$$

$$\lambda = \sqrt{(u_{20} - u_{02})^2 + 4u_{11}^2}$$

Outra forma de se obter a excentricidade da elipse do conjunto de pixels da imagem, e que foi utilizada nesta

implementação, é por meio da divisão do maior autovetor (λ_{max}) pelo menor autovetor (λ_{min}) da matriz de covariância [5] que descreve a distribuição dos pontos bidimensionais (coordenadas x e y) que descrevem o objeto de interesse. Os autovetores dessa matriz de duas dimensões podem ser obtidos pelas fórmulas:

$$\begin{aligned}\lambda_{max} &= \frac{(u_{20} + u_{02}) + \sqrt{(u_{20} - u_{02})^2 + 4u_{11}^2}}{2} \\ \lambda_{min} &= \frac{(u_{20} + u_{02}) - \sqrt{(u_{20} - u_{02})^2 + 4u_{11}^2}}{2}\end{aligned}\quad (9)$$

Por fim, também é possível se obter através dos momentos da imagem informações relacionadas com a correlação entre as coordenadas x e y de cada ponto pertencente ao objeto. Uma regressão linear simples é uma extensão do conceito de correlação/covariância. Ela tenta explicar uma variável, a qual é chamada variável dependente, usando a outra variável, chamada variável independente. Se as duas variáveis são plotadas uma contra a outra num gráfico de espalhamento, considerando Y como a variável dependente e X a independente, com Y no eixo vertical e X no eixo horizontal, a regressão tenta ajustar uma linha reta através dos pontos de tal modo que minimiza a soma dos desvios quadrados dos pontos da linha. Conseqüentemente, ela é chamada de regressão ordinária dos mínimos quadrados. Quando tal linha é ajustada, dois parâmetros emergem - um é o ponto em que a linha corta o eixo Y , chamado de intercepção da regressão, e o outro é a inclinação da linha de regressão.

Em momentos da imagem, a covariância da imagem é obtida através do momento central u_{11} e, como já dito, u_{20} e u_{02} as variâncias em X e Y , respectivamente. A inclinação da regressão é obtida da divisão entre a covariância em X e Y sobre a variância em X [9]. O momento central u_{11} se refere à covariância dos pixels em X e Y .

Portanto, a inclinação da regressão (denominada b) é calculada através da fórmula:

$$b = \frac{cov_{xy}}{var_x} = \frac{u_{11}}{u_{20}} \quad (10)$$

Enquanto a intercepção da regressão (aqui denominada a) é a média em Y (\bar{y}) subtraindo a multiplicação de b (inclinação) com a média em X (\bar{x}). Desta forma, a fórmula para o cálculo é:

$$a = \bar{y} - b\bar{x} = y_c - bx_c \quad (11)$$

4.1 Momentos de Hu

Um dos maiores obstáculos para o reconhecimento de padrões é a falta de invariância das características. Um objeto que por algum motivo tenha sido transladado, rotacionado, ou teve sua escala modificada, deve ser reconhecido

da mesma forma pelo sistema classificador. Ou seja, as características extraídas da imagem não podem variar devido a estas alterações. Por exemplo, em uma aplicação de reconhecimento de faces, a Figura 1 abaixo deve apresentar as características relevantes para sua discriminação com os mesmos valores para as duas imagens, mesmo que uma esteja um pouco rotacionada em relação à outra.



Figura 1. Imagens da mesma face rotacionada

Como citado em [7], o primeiro trabalho de renome empregando momentos para processamento de imagens e reconhecimento de padrões foi desenvolvido por Hu e Alt. Baseado na teoria de invariâncias algébricas, Hu derivou combinações relativas e absolutas de momentos que são invariantes à escala, à posição e à orientação. Para os momentos de segunda e terceira ordem, temos os conhecidos sete momentos de Hu, que são invariantes à escala, rotação e translação, ou seja, os valores desses momentos em uma imagem original são os mesmos apresentados se ela for rotacionada, transladada ou tiver sua escala alterada. Esses momentos são definidos em [2] *apud* [4] pelas seguintes fórmulas:

$$\begin{aligned}\varphi_1 &= n_{20} + n_{02} \\ \varphi_2 &= (n_{20} - n_{02})^2 + 4n_{11}^2 \\ \varphi_3 &= (n_{30} - 3n_{12})^2 + (3n_{21} - n_{03})^2 \\ \varphi_4 &= (n_{30} + n_{12})^2 + (n_{21} + n_{03})^2 \\ \varphi_5 &= (n_{30} - 3n_{12})(n_{30} - n_{12}) \\ &\quad ((n_{30} + n_{12})^2 - (n_{21} + n_{03})^2) \\ &\quad + (3n_{21} - n_{03})(n_{21} + n_{03}) \\ \varphi_6 &= (n_{20} - n_{02}) \\ &\quad ((n_{30} + n_{12})^2 - (n_{21} + n_{03})^2) \\ &\quad + 4n_{11}(n_{30} + n_{12})(n_{21} + n_{03}) \\ \varphi_7 &= (3n_{21} - n_{03})(n_{30} + n_{12}) \\ &\quad ((n_{30} + n_{12})^2 - 3(n_{21} + n_{03})^2) \\ &\quad - (n_{30} - 3n_{12})(n_{21} + n_{03}) \\ &\quad (3(n_{30} + n_{12})^2 - (n_{21} + n_{03})^2)\end{aligned}\quad (12)$$

5 Implementação

Durante o estudo das técnicas de extração de atributos através de momentos estatísticos foi desenvolvida uma implementação prática, compreendendo os conceitos de momentos e suas funções. Para realizar esta implementação foi utilizado o ImageJ, que é um pacote para processamento de imagens de código-fonte aberto³. O ImageJ fornece a opção de implementação de plugins, que são classes em Java onde se implementam as especificações necessárias, que serão compiladas para identificação do ImageJ.

O plugin desenvolvido recebe como entrada uma imagem digital qualquer em que se deseja extrair suas características. Primeiramente é gerada uma réplica desta entrada, para que a imagem original não sofra alterações, e os procedimentos passam a ser executados na cópia da imagem.

Esta réplica é submetida a um processo de binarização (os valores dos pixels da imagem podem possuir apenas dois valores: 0 ou 1; onde se associam as cores preto e branco para sua representação), que também é executado por um plugin do ImageJ (*Threshold*). Juntamente com a binarização ocorre a segmentação do objeto de interesse. Para permitir a obtenção de características locais do objeto de interesse, é possível dividir a imagem em regiões, como sugerido em [3], para que se possa calcular os momentos da imagem não apenas globalmente, mas também para cada uma das regiões. Na implementação atual, a imagem é dividida em quatro partes iguais, como mostra a Figura 2⁴.

Para a realização da extração das características da imagem foram utilizados, neste trabalho, os momentos regulares de ordem 0 e 1 (m_{00} , m_{10} , m_{01}), e os momentos centrais de ordem 2 (u_{11} , u_{20} , u_{02}). Para cada uma das regiões são calculadas cinco características: os respectivos centros de massa e os desvios padrão nos eixos X e Y e o total de pixels.

À medida que os momentos regulares das regiões são calculados eles são acumulados nos momentos regulares da imagem como um todo, para evitar repetições de códigos desnecessárias, melhorando o desempenho do algoritmo. Após calculados os atributos de cada região da imagem é calculado o centro de massa da imagem inteira, e com isto, são calculados os momentos centrais da imagem. Do objeto completo são extraídas as seguintes características: total de pixels, centros de massa e desvios padrão em X e Y, excentricidade, orientação, o valor do intercepto e o coeficiente angular (inclinação) da regressão linear; ressaltando que todas os valores destas características são obtidos através dos

³O ImageJ foi desenvolvido em Java e inspirado no NIH Image do Macintosh.

⁴O ImageJ possui um recurso denominado ROI (Region Of Interest), que permite a seleção da região que é de interesse dentro da imagem. Dentro do plugin, todo o processamento é realizado sobre a ROI e não sobre a imagem completa.

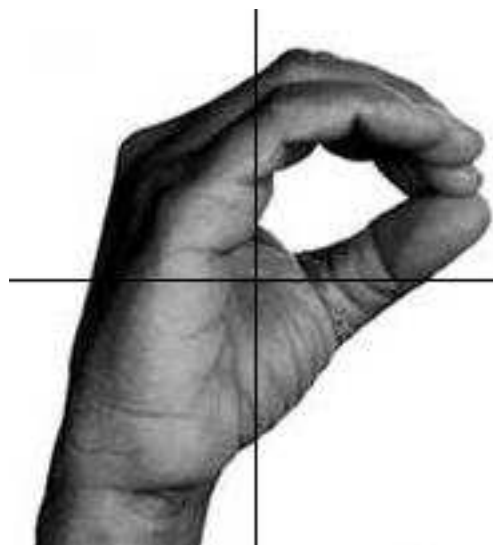


Figura 2. Imagem de entrada dividida em quatro partes iguais

momentos estatísticos.

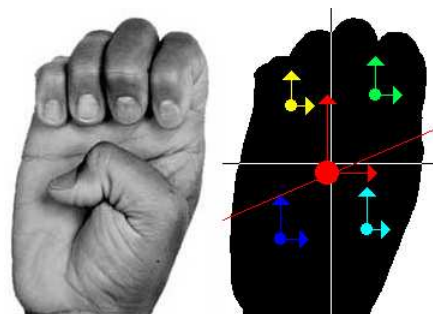


Figura 3. Letra E em LIBRAS

Os resultados dos cálculos do plugin podem ser visualizados pelo usuário de duas formas: textual e/ou graficamente. Na forma textual, as características exibidas da imagem inteira são: o centróide, o desvio padrão, o ângulo de orientação e o valor da excentricidade; e das regiões são os respectivos centróides e os desvios padrão. De forma gráfica as informações possuem cores, para identificação da região a que pertencem. A região superior-esquerda possui a cor amarela; a superior-direita, verde; a inferior-esquerda, azul escuro; a inferior-direita, azul claro; e a imagem inteira, vermelho. As regiões são evidenciadas com linhas delimitando suas fronteiras. O centro de cada região é representado por um círculo com a cor respectiva da região. Os desvios padrão das regiões (inclusive de toda a imagem) são representados em forma de vetores, sendo o da vertical referente ao desvio no eixo Y, e o horizontal ao eixo

X. A regressão linear é representada por uma linha vermelha, que corta a imagem. A Figura 3 mostra o resultado da aplicação do plugin em uma imagem de uma mão realizando uma configuração de mão da Língua Brasileira de Sinais, LIBRAS.

6 Experimentos

O plugin desenvolvido é de fácil utilização; basta apenas selecioná-lo nas opções de plugins do ImageJ e aguardar seu processamento. Não há a necessidade de mudar o tipo da imagem, nem binarizá-la, pois o mesmo já é feito automaticamente em sua execução. Durante a fase de testes deste trabalho foram capturadas imagens do site <http://www.libras.org.br/> que aborda o assunto da Linguagem Brasileira de Sinais.

Os resultados gráficos da execução do extrator nas imagens podem ser observados através das ilustrações abaixo, em que a Figura 9 é uma imagem gerada manualmente para os testes no aplicativo, e as Figuras 4, 5, 6, 7 e 8 representam nesta ordem os gestos utilizados em LIBRAS para simbolizar os números 0, 2 e 5, e as letras A e T do alfabeto.

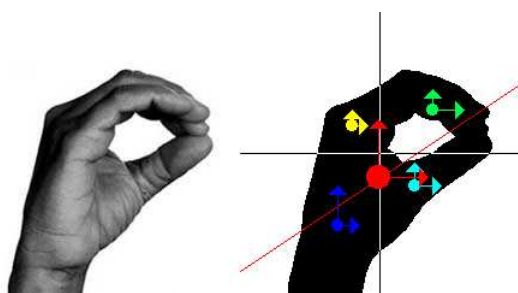


Figura 4. Símbolo 0 (zero) em LIBRAS

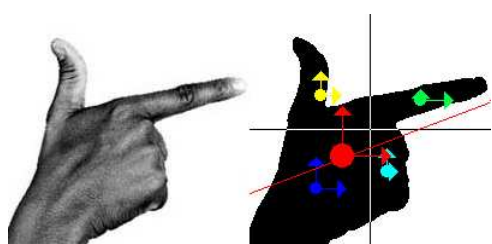


Figura 5. Símbolo 2 (dois) em LIBRAS

Um exemplo do benefício obtido pela divisão da imagem em quatro partes é demonstrado na Figura 10, em que são representadas as imagens das letras I (esquerda) e S (direita), no alfabeto LIBRAS. Visualizando as duas imagens como um todo percebe-se que elas são relativamente parecidas. Por este motivo algumas características globais ob-

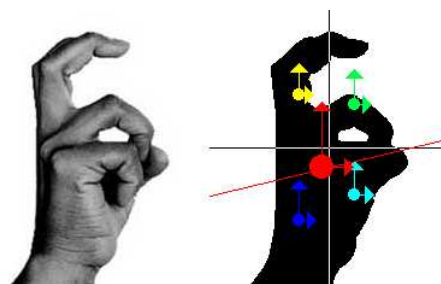


Figura 6. Símbolo 5 (cinco) em LIBRAS

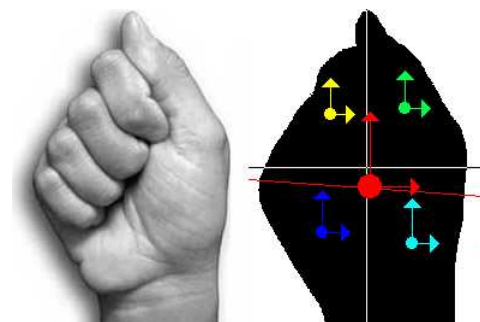


Figura 7. Letra A em LIBRAS

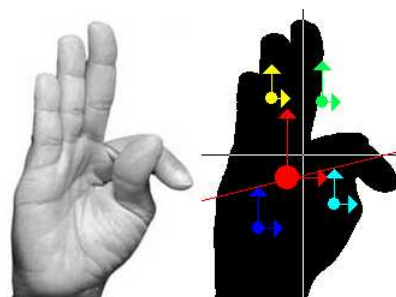


Figura 8. Letra T em LIBRAS

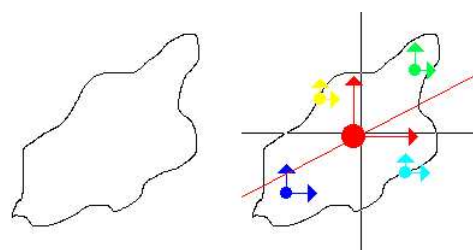


Figura 9. Imagem criada manualmente para teste

tidas das imagens foram próximas, mesmo sendo imagens de classes diferentes. Este tipo de problema pode ocasionar a classificação incorreta da imagem, comprometendo a

eficácia do sistema classificador.

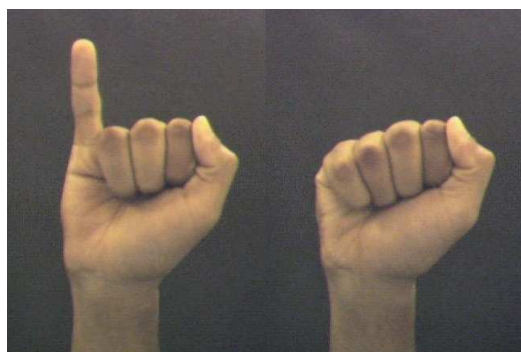


Figura 10. Letras I (esquerda) e S (direita) em LIBRAS

Os valores apresentados na Tabela 1 são algumas características globais que foram extraídas de ambas as imagens.

Características	Letra S	Letra I
Total de Pixels	46434	51360
Centro de Massa (X)	153	147
Centro de Massa (Y)	269	253
Desvio Padrão (X)	54	54
Desvio Padrão (Y)	78	90

Tabela 1. Resultados obtidos da extração de características da imagem inteira

É possível analisar uma variação pequena das características de uma imagem para outra, o que em um sistema de classificação poderia ser identificado como entradas pertencentes à mesma classe, o que não é verídico. As informações extraídas de cada parte da imagem auxiliam a discriminação correta de cada entrada. As características de cada quadrante da imagem são exibidas na tabela 2, sendo que a região 1 corresponde ao canto superior-esquerdo, a região 2 ao canto superior-direito, a região 3 ao canto inferior-esquerdo e a região 4 ao canto inferior-direito.

No caso das letras I e S o quadrante que melhor auxiliou a discriminação foi o quadrante 1, pois apresentou maior variação nos valores das características. Esta variação pode ser melhor percebida de forma gráfica, como mostra a Figura 11.

7 Conclusão

A extração de características através dos momentos da imagem é uma técnica muito eficaz para o reconhecimento de padrões e muito útil para o cálculo de valores estatísticos,

Características	Letra S	Letra I
Região 1		
Total de Pixels	5757	10706
Centro de Massa (X)	116	200
Centro de Massa (Y)	187	285
Desvio Padrão (X)	28	29
Desvio Padrão (Y)	18	54
Região 2		
Total de Pixels	8322	8551
Centro de Massa (X)	215	215
Centro de Massa (Y)	179	178
Desvio Padrão (X)	31	31
Desvio Padrão (Y)	22	22
Região 3		
Total de Pixels	20778	20866
Centro de Massa (X)	113	113
Centro de Massa (Y)	319	320
Desvio Padrão (X)	28	28
Desvio Padrão (Y)	62	62
Região 4		
Total de Pixels	11577	11237
Centro de Massa (X)	200	198
Centro de Massa (Y)	285	285
Desvio Padrão (X)	29	28
Desvio Padrão (Y)	54	54

Tabela 2. Resultados obtidos da extração de características de cada região da imagem

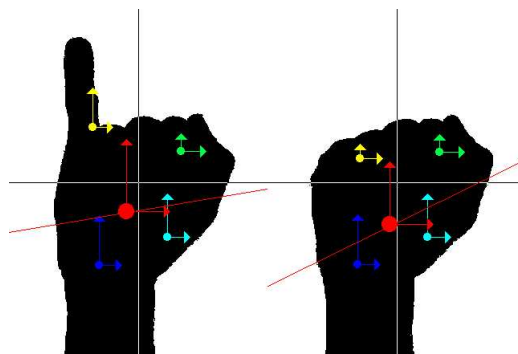


Figura 11. Letras I (esquerda) e S (direita) em LIBRAS após o processo de extração de características

reutilizando valores calculados anteriormente, o que auxilia nos cálculos e diminui o tempo de processamento destes valores. Ao substituir por este trabalho o módulo que realizava a extração de características da imagem dentro do projeto SIGUS, que fora comentado na seção 1, foi possível verificar um ganho considerável em velocidade de proces-

samento, o que possibilitou uma representação das imagens de forma mais rápida ao usuário, não retardando o aparecimento dos quadros.

Os próximos trabalhos a serem desenvolvidos com base neste projeto é a seleção das características e a implementação destas técnicas de extração e visualização de características da imagem no projeto SIGUS, para melhora do desempenho e eficácia atual. Parte deste trabalho já foi transferida ao projeto SIGUS, faltando ainda a implementação da visualização gráfica em tempo real das imagens capturadas.

8 Agradecimentos

Este trabalho recebeu apoio financeiro da Universidade Católica Dom Bosco, UCDB, e da Fundação de Apoio ao Desenvolvimento do Ensino, Ciência e Tecnologia do Estado de Mato Grosso do Sul, FUNDECT. O acadêmico Kleber Padovani Souza é bolsista de iniciação científica PIBIC/CNPq.

Referências

- [1] L. S. Alves. Momentos estatísticos de palavras manuscritas. In *XII Seminário de Iniciação Científica da PUCPR*, Novembro 2004.
- [2] G. C. Chavez and Z. Lhiang. Sistema celular para reconhecimento de padrão invariante. In *IV Workshop em Tratamento de Imagens*, pages 2–3, 2003.
- [3] W. T. Freeman, D. B. Anderson, P. A. Beardsley, C. N. Dodge, M. Roth, C. D. Weissman, W. S. Yerazunis, H. Kage, K. Kyuma, Y. Miyake, and K. ichi Tanaka. Computer vision for interactive computer graphics. *IEEE Computer Graphics and Applications*, 18(3):42–53, 1998.
- [4] M. Hu. Pattern recognition by invariant moments. *IRE Transactions on Information Theory*, pages 179–187, 1961.
- [5] J. Scharcanski and C. T. Dodson. Stochastic texture estimators for local anisotropy and its variability. *IEEE Transactions on Instrumentation and Measurement*, 49:4–5, 2000.
- [6] A. Khotanzad and Y. H. Hong. Invariant image recognition by zernike moments. *IEEE Transactions on Pattern analysis and machine intelligence*, 12:1, 1990.
- [7] S. X. Liao. *Image Analysis by Moments*. PhD thesis, Department of Electrical and Computer Engineering, University of Manitoba, Winnipeg, Manitoba, Canada, 1993.
- [8] H. Pistori and M. C. Pereira. Integração dos ambientes livres weka e imagej na construção de interfaces guiadas por sinais visuais. In *Anais do V Workshop de Software Livre - WSL*, Porto Alegre, RS, 2-5 de Junho 2004.
- [9] A. Sartoris. *Estatística e Introdução à Econometria*. Saraiva, 2003.
- [10] S. Theodoridis and K. Koutroumbas. *Pattern recognition*. Academic press, 1999.
- [11] E. Vidal, A. Juan, D. Picó, and R. Paredes. Aprendizaje y percepción - tema 2: Preproceso y extracción de características imágenes. Technical report, Facultad de

Informática, Universidad Politécnica de Valencia, febrero 2005.