

COMPARAÇÃO DE HISTOGRAMAS DE IMAGENS DIGITAIS PARA DETERMINAÇÃO DE SIMILARIDADE EM SEMENTES DE MILHO

Jonathan Rui de Lima (Departamento de Informática - UEPG) E-mail: rlima.jonathan@hotmail.com

Márcio Hosoya Name (Gestão de Tecnologia da Informação - UFPR) E-mail: name@ufpr.br

Maria Salete Marcon Gomes Vaz (Departamento de Informática – UEPG) E-mail: salete@uepg.br

Luciane Henneberg (Departamento de Fitotecnia e Fitossanidade – UEPG) E-mail: lucihenne@gmail.com

David de Souza Jaccoud Filho (Departamento de Fitotecnia e Fitossanidade – UEPG) dj1002@uepg.br

Rosane Falate (Departamento de Informática – UEPG) E-mail: rfalate@yahoo.com

Resumo: Este trabalho apresenta um *Software* de comparação de imagens pelo uso da métrica de interseção nos histogramas destas imagens. Para os testes iniciais, buscando a aplicabilidade de tal tecnologia no agronegócio, foram determinadas similaridades entre sementes de milho. Foram obtidos como resultados o valor de 1,00000, quando foram comparados os histogramas de uma mesma imagem, e o valor de 0,680135, para o caso de maior dissimilaridade, que foi entre uma imagem base e a de uma semente de milho diferente, respectivamente, assinalando para o possível uso deste método, na determinação de similaridades. Dessa forma, esta tecnologia aparentemente pode ser usada em análises da área agrícola como: seleção de sementes, teste de vigor de plântulas, determinação de textura, danos mecânicos, entre outras atividades, que demandam reconhecimento de padrões, por um técnico, de forma visual. Uma vez que este *software* foi realizado em Linguagem Java[®], destaca-se ainda o uso deste em varias plataformas, ampliando sua aplicabilidade para auxílio na área de agronegócios.

Palavras-chave: processamento digital de imagens, interseção, visão computacional.

COMPARISON OF DIGITAL IMAGE HISTOGRAMS FOR DETERMINATION OF SIMILARITY IN CORN SEEDS

Abstract: This paper presents image comparison software by the use of the intersection metric in the histograms of images. For the initial tests, looking for the applicability of such technology in agribusiness, we choose determine similarities between corn seeds. The obtained results varied from 1.00000, when comparing the histograms of the same image, to 0.680135 in case of greater dissimilarity, between a base image and a different seed corn, respectively, pointing to the possible use of this method in determining similarities. So apparently this technology can be used in agricultural procedures as seed selection, test seedling vigor, determination texture, mechanical damage, and other activities that require a technician for pattern recognition in a visual way. Furthermore, this software was done in Java[®], so it can be used on multiple platforms, expanding its applicability in the agricultural area.

Keywords: digital image processing, intersection, computer vision.

1. INTRODUÇÃO

A visão computacional e o processamento digital de imagens vêm sendo amplamente utilizadas nas mais diversas áreas do conhecimento: astronomia, medicina, análise de impressões digitais, reconhecimento de assinaturas, sensoriamento remoto, multimídia, manufatura, robótica e sistemas produtivos (Rudek, Coelho e Canciglieri, 2001). Isso porque a visão computacional procura coletar informações do mundo real, a partir de imagens, e que sejam interessantes e desejadas por uma determinada área de conhecimento (CALVO, 2011).

A comparação de histogramas é uma das maneiras de se obter informações das imagens (HUET e HANCOCK, 1996; HISTOGRAM COMPARISON, 2012). Um histograma é definido como a quantificação do número de pixels para cada valor de intensidade de uma imagem ou uma representação gráfica da distribuição de intensidade ou (GONZALEZ e WOODS, 2002).

Este trabalho apresenta um *software* para busca de similaridades de imagens. Para isso, foi adotada a métrica da interseção de comparação de histogramas. Pelas possibilidades de aplicação no agronegócio na região dos Campos Gerais, para os testes com o *software* foram usadas sementes de milho.

2. METODOLOGIA

2.1 TECNOLOGIAS UTILIZADAS

Para o processamento digital de imagens e aplicação de visão computacional foi usada a biblioteca OpenCV (*Open Source Computer Vision – Visão Computacional de Código Aberto*) (MARENGONI E STRINGHINI, 2009). Essa biblioteca foi desenvolvida pela Intel, escrita em Linguagem C/C++, e atualmente dá suporte a Linguagem Python (HEIDEMANN, 2011). Entretanto, para que suas funções fossem acessadas por códigos em Linguagem Java foi necessário o uso de um *plugin*: o JavaCV (AUDET, 2010). Esse *plugin* implementa várias funções já existentes no OpenCV, as quais são acessadas via JNI (*Java Native Interface – interface nativa Java*).

Segundo Alves (2004), JNI é uma interface incluída na JDK (*Java Development Toolkit*), para definir métodos nativos e embutir a Máquina Virtual Java, dentro de aplicações nativas. Métodos nativos são funções escritas, em linguagens como C, C++ e até mesmo Assembly, compiladas em DLLs (Windows) ou Sistemas Operacionais (Linux e Solaris). JNI serve como uma ponte entre Java e a parte nativa de uma aplicação, provendo funcionalidades para comunicação entre as partes. Também, estão incluídas na JDK ferramentas para mapear protótipos de métodos e tipos de dados de Java para C / C++.

2.2 MATERIAIS

Para realização do experimento, sementes de milho de tamanhos distintos foram fornecidas, pela Fazenda Escola, da Universidade Estadual de Ponta Grossa. De posse das sementes, cada uma delas foi digitalizada, com uma multifuncional Epson Stylus® TX115, resolução de 600 dpi, RGB (*Red, Green, Blue – vermelho, verde, azul*) e em modo profissional. A opção de utilizar o modo profissional foi para que todas as imagens tivessem a mesma quantidade de pixels (425 x 261). As imagens foram salvas em JPEG.

Neste trabalho foram usadas quatro imagens de sementes. A partir dessas imagens, iniciaram-se os testes do software, para comparação de histogramas, através da métrica de Interseção. O sistema foi desenvolvido utilizando as plataformas livres Netbeans 7.1.2 da Oracle, OpenCV Versão 2.4.0, e, ainda, javacv 0.1 (*plugin Java para OpenCV*), executado em um Processador Intel i5 2.40 GHz, 4,0 GBs de memória RAM e Sistema Operacional Windows 2008 Server R2.

2.3 MÉTODOS

A métrica de comparação de histograma por interseção é dada pela equação (HISTOGRAM COMPARISON, 2012):

$$d(H_1, H_2) = \sum_I \min(H_1(I), H_2(I))$$

onde H corresponde ao histograma da imagem e I é o número total de cores, no espaço de cor específico da imagem.

Para ser aplicada a essa métrica, observou-se que são necessários primeiro os procedimentos: passagem das informações das imagens para variáveis do programa; conversão do espaço de cor da imagem RGB para HSV (*Hue, Saturation and Value/Luminance* – matiz, saturação e valor/luminância), a maneira mais natural para descrever o conjunto de cores da imagem; e normalização. Assim, o algoritmo montado para o *software* implementou, primeiramente, essas etapas e depois foi aplicada a métrica selecionada neste trabalho.

Construído o sistema com os algoritmos indicados e de posse das imagens, selecionou-se a imagem de referência, a imagem de comparação e utilizou-se a métrica para todas as imagens, sempre em comparação com a imagem de referência.

3. RESULTADOS E DISCUSSÃO

Para aplicação da métrica de interseção, com o objetivo de definir o nível de similaridade entre sementes, foram escolhidas quatro sementes para teste, na seguinte ordem: Imagem base chamada de *imgBase* e três imagens a serem comparadas com a primeira, denominadas, *imgComp1*, *imgComp2* e *imgComp3*.

O experimento foi conduzido pelo cruzamento entre o histograma retirado da imagem base x imagens de comparação, seguindo a sequência: *imgBase* x *imgBase*, *imgBase* x *imgComp1*, *imgBase* x *imgComp2* e *imgBase* x *imgComp3*. O valor de referência foi retirado da comparação entre a *imgBase* VS *imgBase*, e os outros valores resultantes podem definir o quanto é similar as outras imagens com a imagem base.

3.1 SOFTWARE CHI (COMPARAÇÃO DE HISTOGRAMA POR INTERSEÇÃO)

A seguir é apresentado o software que trata as imagens desde seu carregamento até o resultado da comparação entre elas.

Os histogramas são sempre carregados dois a dois, ou seja, compara-se sempre em pares, o que possibilitará comparação precisa de um número grande de imagens, sendo útil em testes que objetivam descobrir “defeitos” em lotes de sementes, a partir de uma imagem base considerada adequada.

Os comandos para a passagem das informações de duas imagens, a de referência (*imgBase*) e a de comparação (*imgComp*) para suas respectivas variáveis, estão mostrados na Figura 1.

```
//OBTENÇÃO DAS IMAGENS
CvHistogram h = img_base(imgBase);
CvHistogram h1 = img_base(imgComp);
String[] retorno;
```

Figura 1: Obtenção da Imagem

Foram criados dois objetos da classe *CvHistogram*, passando como parâmetro as imagens. A partir dessas instâncias, tem-se os objetos *h* (objeto da *imgBase*) e *h1* (objeto da *imgComp*). Como apresentado anteriormente, as imagens são comparadas em pares. Portanto, para cada *imgComp* (*imgComp1*, *imgComp2* e *imgComp3*), são criados objetos distintos para comparar com *imgBase*.

A sequência de comandos, para a conversão de cor e normalização, é apresentada na Figura 2.

```
public CvHistogram img_base(String var) {
    CvHistogram hist = null;
    IplImage src = cvLoadImage(var);

    IplImage hsv = cvCreateImage(cvGetSize(src), 8, 3);
    cvCvtColor(src, hsv, CV_BGR2HSV);

    IplImage h_plane = cvCreateImage(cvGetSize(src), 8, 1);
    IplImage s_plane = cvCreateImage(cvGetSize(src), 8, 1);
    IplImage v_plane = cvCreateImage(cvGetSize(src), 8, 1);

    cvSplit(hsv, h_plane, s_plane, v_plane, null); //cvCvtPixToPlane

    int h_bins = 50;
    int s_bins = 60;

    {
        int hist_size[] = {h_bins, s_bins};
        float h_ranges[] = {0, 180};
        float s_ranges[] = {0, 255};

        float[][] ranges = {h_ranges, s_ranges};

        hist = cvCreateHist(hist_size.length, hist_size, CV_HIST_ARRAY, ranges, 1);
    }
    IplImage planes[] = {h_plane, s_plane, v_plane};
    cvCalcHist(planes, hist, 0, null);
    cvNormalizeHist(hist, 1);
    return hist;
}
```

Figura 2: Conversão de Cor e Normalização

Por fim, com os histogramas de cada imagem criados e devidamente alocados nos objetos `h` e `h1`, é chamado o método de comparação por interseção, referenciado pelo valor 2 que está atribuído em `compMet` (método de comparação). A Figura 3 demonstra o retorno do valor da comparação entre a imagem base (`imgBase`) e a imagem de Comparação (`imgComp`) e, também, o valor de comparação entre os histogramas da imagem base, para valor de referência.

```
if (compMet == 0) {
    retorno = new String[2];
    retorno[0] = String.valueOf(cvCompareHist(h, h1, 0));
    retorno[1] = String.valueOf(cvCompareHist(h, h, 0));
} else if (compMet == 1) {
    retorno = new String[2];
    retorno[0] = String.valueOf(cvCompareHist(h, h1, 1));
    retorno[1] = String.valueOf(cvCompareHist(h, h, 1));
} else if (compMet == 2) { //MÉTODO DE INTERSEÇÃO UTILIZADO
    retorno = new String[2];
    retorno[0] = String.valueOf(cvCompareHist(h, h1, 2)); //COMPARAÇÃO DO HISTOGRAMA ATRAVÉS
    retorno[1] = String.valueOf(cvCompareHist(h, h, 2)); //DO MÉTODO INTERSEÇÃO IMPLEMENTADO PELA BIBLIOTECA OPEN CV
} else if (compMet == 3) {
    retorno = new String[2];
    retorno[0] = String.valueOf(cvCompareHist(h, h1, 3));
    retorno[1] = String.valueOf(cvCompareHist(h, h, 3));
} else {
    retorno = new String[2];
    retorno[0] = "Imagem ou Método Incorreto";
}
}
```

Figura 3: Comparação dos Histogramas

A Figura 4 mostra a interface do Software CHI, a sequência para utilização do sistema é dada com os seguintes passos: 1) Carregar as imagens *imgBase*, *imgComp1*, *imgComp2* e *imgComp3*; 2) Clicar botão de comparação, o qual irá acionar os comandos apontados nas Figuras 1 a 3; e 3) Retornar e exibir os valores das comparações, nos componentes *vB* (valor comparação base, *imgBase* x *imgBase*), *vC1* (valor comparação 1, *imgComp1* x *imgBase*), *vC2* (valor comparação 2, *imgComp2* x *imgBase*) e *vC3* (valor comparação 3, *imgComp3* x *imgBase*).

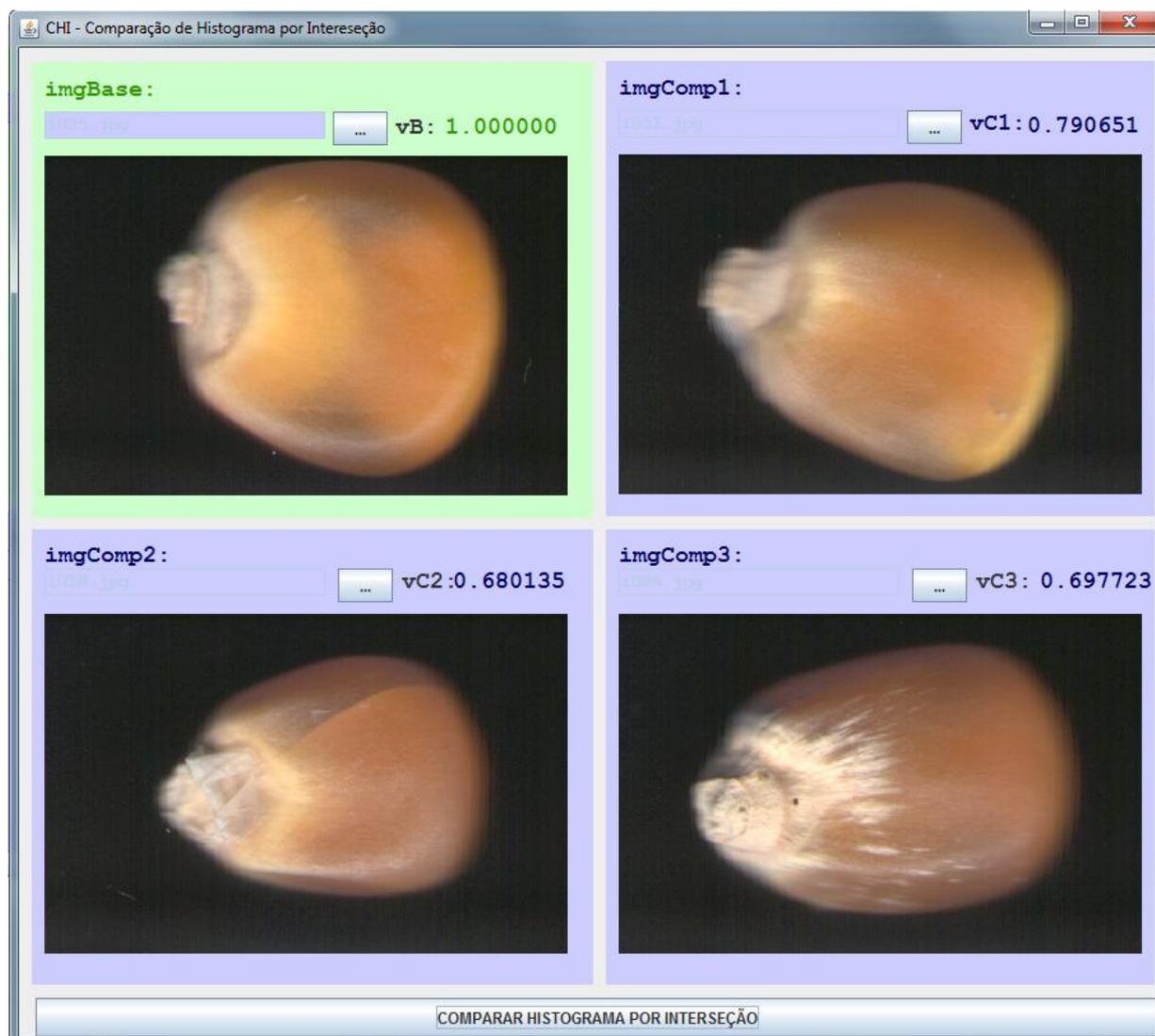


Figura 4 – Sistema desenvolvido para comparação de histogramas por interseção.

Como pode ser observado da Figura 4, ao se aplicar o sistema de detecção de similaridades por interseção, ele retorna o valor de 1,000000, quando é feita a comparação da imagem de uma determinada semente com ela mesma (*imgBase* x *imgBase*). Por outro lado, ao comparar sementes diferentes, obtiveram-se valores menores que 1,000000, sendo que a maior diferença foi de 0,680135, entre a *imgBase* e *imgComp2*.

4. CONCLUSÕES

Foi desenvolvido um *software* para detecção de similaridade entre sementes de milho, através da utilização da métrica de interseção. Todas as sementes foram distintas usando o método proposto.

A vantagem de se usar um programa computacional para essa finalidade é que ele é não destrutivo, usa de imagens das sementes para avaliar a similaridade, e possui uma avaliação objetiva, diferente da humana, que é subjetiva. Logo, os resultados podem variar ao longo do tempo e do avaliador.

A partir dos resultados encontrados, acredita-se que o sistema desenvolvido, ou uma variação dele, pode ajudar nos processos de inspeção visual, comumente usados na agricultura, como uma ferramenta auxiliar de avaliação.

AGRADECIMENTOS

Os autores agradecem o Departamento de Fitopatologia por ceder o material para aquisição de imagens.

REFERÊNCIAS

ANDREOLI, C.V; PIGORINE, E.S; HOPPEN C; TAMANINI, C.R; NEVES, P.S. *Produção, composição e constituição de lodo de estação de tratamento de água.* In: ANDREOLI, C.V (coord.). *Alternativas de Uso de Resíduos de Saneamento.* Rio de Janeiro: ABES, p.29- 48, 2006.

ALVES, G. D. *Trabalho de Formatura Supervisionado.* 2004. Disponível em: <<http://www.linux.ime.usp.br/~cef/mac499-04/monografias/givanildo/jni.html>>. Acesso em 31 Jul 2012.

AUDET, S. *JavaCV.* 2010. Disponível em <<http://code.google.com/p/javacv/>> . Acesso em 30 Jul 2012.

Calvo, A. B. *Extraccion de Caracteristicas para Recuperacion de Imagenes Basada em Contenido.* Universidad Rey Juan Carlos. 2011.

GONZALEZ, RAFAEL C; WOODS, RICHARD E. *Digital Image Processing,* 2002, Prentice-Hall, Inc.

HEIDEMANN, G. Z. *Aplicando Diversas Técnicas de Realidade Aumentada no Desenvolvimento de um Jogo.* Universidade do Vale do Itajaí. Itajaí, SC. 2011.

HISTOGRAM COMPARISON - OPENCV V2.4.0 DOCUMENTATION. Disponível em: <http://docs.opencv.org/2.4/doc/tutorials/imgproc/histograms/histogram_comparison/histogram_comparison.html>. Acesso em: 30 jul 2012.

HUET, B., HANCOCK, E. R. *Structural indexing of infra-red images using statistical histogram comparison.* In Proc. 3rd Int. Workshop Image and Signal Processing, Manchester, U.K., Nov. 1996, pp. 653–656.

MARENGONI, M.; STRINGHINI, S. *Revista de Informática Teórica e Aplicada.* v. 16, n. 1, 2009.

RUDEK, M.; COELHO, L. S.; CANGIHLIERI J. O. *Visão Computacional Aplicada a Sistemas Produtivos: Fundamentos e Estudo de Caso.* In: XXI ENCONTRO NACIONAL DE ENGENHARIA DE PRODUÇÃO, 2001, Salvador. XXI Encontro Nacional de Engenharia de Produção. 2001. |