

# **DIVMAGE: Componente de software capaz de manter o direito autoral sobre imagens na web**

Leonardo Rossi Rodrigues, Flávio Luiz Schiavoni  
Departamento de Informática – Universidade Estadual de Maringá (UEM)  
Maringá – PR – Brasil

rossileonardo@gmail.com, fls@ime.usp.br

**Resumo.** Ao disponibilizar uma imagem na Internet, outras pessoas podem copiá-la e reproduzi-la sem que o proprietário o saiba. Este trabalho propõe um modelo para representação de imagem na Internet que impede a cópia do arquivo. Este componente chamado DIVMAGE representa uma imagem via *browser* evitando cópias indevidas e reproduções, permitindo com que o autor fique despreocupado em divulgar suas imagens. Toda imagem armazenada na pasta específica é convertida automaticamente em tempo de execução no modelo de representação, o qual exibe a imagem normalmente, mas o usuário não consegue salvar a imagem da página.

**Palavras-chaves.** Computação Gráfica, Direito Autoral, Processamento de Imagem

*Abstract.* When images are provided on the Internet, others may copy and reproduce it without the owner's previous knowledge. This paper proposes a model for image representation on the Internet that prevents file copying. This component named DIVMAGE represents an image via browser avoiding unauthorized copying and reproducing, allowing authors to feel protected when disclosing their images. Any image stored in this specific file is automatically converted in runtime in the representation model, which displays the image normally, but prevents the user to save the image on the page.

**Keywords:** Computer Graphics, Copyright, Image Processing

## **1. Introdução**

Antigamente os editores e autores não reivindicavam um direito, mas apenas uma garantia de sua produção intelectual pela parte dos autores e de estabelecer um sistema de monopólio pelos editores e a previsão de sanções aos infratores desses privilégios, como a apreensão das cópias e pagamento de indenizações (DEGUCHI, 2007). Surge então em 14 de abril de 1710 na Grã-Bretanha, a primeira lei específica sobre o direito de autor com o *Copyright Act*, e em 1790 a Inglaterra começa a reconhecer este mesmo direito (MARTINS, 1998).

Existe uma diferença conceitual entre Direito Autoral e Direito do Autor (ORRICO, 2004), apesar de ambas estarem inseridas no ramo do Direito Privado. BITTAR(2004) define o Direito do Autor como “o ramo de direito privado que regula as relações jurídicas advindas da criação e da utilização econômica de obras intelectuais estéticas, e compreendidas na literatura, nas artes e nas ciências”, e o ASCENSAO(1980) fala que o Direito Autoral “abrange, além disso, os chamados direitos conexos do direito de autor, como os direitos dos artistas intérpretes ou executantes, dos produtores de fonogramas e dos organismos de radiodifusão”.

Os objetos do direito autoral são as obras intelectuais de qualquer modo exteriorizadas, ou seja, protege-se as formas de expressão das ideias e não as ideias em si. Segundo GANDELMAN (2007), “O sujeito do direito autoral é, portanto, o autor, ou ainda o titular de autoria de obra intelectual; o objeto desse direito é a proteção legal da própria obra criada e fixada em qualquer

suporte físico, ou veículo material. ”

Existem dois elementos do direito autoral (PEREIRA, 2003): Direito moral: garantem ao criador reivindicar a autoria da obra, bem como a menção do seu nome na divulgação da mesma e assegurar a integridade da obra, em sua reputação ou honra, além dos direitos de modificá-la ou retirá-la de circulação. Esses direitos são inalienáveis e irrenunciáveis. Direito patrimonial: asseguram ao criador o retorno financeiro de todas as relações econômicas que tenham por objeto a sua obra intelectual. É negociável total ou parcialmente, por tempo determinado, ou indeterminado.

Ao passar dos anos o direito autoral vem se aprimorando, se fortalecendo e sendo discutido cada vez mais entre os países, no Brasil (MARTINS, 1998), o direito autoral foi regulado pela Lei 5 988 de 14 de dezembro de 1993. A partir de 19 de junho de 1998, entrou em vigor a Lei 9.610 de 19 de fevereiro de 1998, a nova lei dos direitos autorais e no mesmo ano a Lei no 9.609 de 1998 que revoga a Lei no 7.646/87 e passa a tratar dos direitos autorais relacionados aos programas de computador, também denominada de Lei do Software.

O direito autoral não é eterno (PECK, 2009), existe uma reserva legal de tempo que é definida pela lei 9.610/98 onde após está data, as obras se tornam domínio público. No caso autoral é de 70 anos, sendo que exclusivamente para software é de 50 anos.

Nos Estados Unidos após o projeto Gutenberg ([www.promo.net/pg](http://www.promo.net/pg)) colocar a disposição milhares de obras em domínio público, na web, a pedido de Hollywood em 1998 o congresso dos Estados Unidos estendeu os direitos autorais em seu país por mais 20 anos (TANENBAUM, 2003).

Muitas formas de não cumprir os direitos autorais são facilmente encontrados utilizando a Internet, como por exemplo, programas conexões *peer to peer (P2P)*, que possibilitava a conexão direta entre dois computadores ligados à Internet que podem enviar e receber arquivos de qualquer tipo (imagem, música, texto, etc.) (PORTO, 2009).

Neste contexto é focado o direito autoral sobre imagens disponíveis na Internet, onde segundo (MARTINS, 1998) “o mesmo princípio que protege a obra originária também protege os direitos conexos, portanto o uso de imagens e sons também depende da autorização do autor para sua reprodução. O que acontece é que, com a facilidade de manipulação através de programas, é possível modificar uma imagem a tal ponto que se torna quase impossível afirmar, ou mesmo provar, que tal imagem pertença mesmo a seu autor.”.

A realização deste projeto provê uma nova forma de representar imagens em sites, dificultando o plágio e mantendo seus direitos. A maioria dos sites possuem imagens, sendo que a maior parte é utilizada apenas para ilustrações e em alguns sites há a necessidade de divulgar imagens de um trabalho. Supondo que o site seja de um fotógrafo, todas as fotos contidas no site serão de seu direito e quanto melhor a qualidade da imagem, melhor será a divulgação de seu trabalho. Neste caso qualquer pessoa poderá acessar e copiar suas fotos, mesmo que possua direitos autorais sobre elas, nem sempre é possível rastrear quem copiou a imagem e nem como foi utilizada.

## 2. Objetivos

O objetivo geral é desenvolver um modelo que permita publicações de imagens na Internet, mantendo seu direito autoral. Os objetivos específicos:

- Estudar sobre os direitos autorais, no contexto da Internet, focando em relação as imagens;
- Estudar sobre formatos existentes de imagens;
- Estudar sobre compactação de imagens;
- Funcionalidades dos filtros existentes no servidor *Tomcat* (servidor *web* Java).;
- Estudar Java2D;
- Estudar funcionalidades do *JavaScript* e HTML;

### 3. Metodologia

Foi utilizado a linguagem Java para o desenvolvimento, utilizando recursos de filtro do *Tomcat* e *javascript*. DIVMAGE é um modelo de representação onde a partir da configuração do filtro do *Tomcat*, as imagens contidas no servidor ao serem solicitadas pela página, serão convertidas em tempo de execução para o formato utilizado.

A primeira fase foi desenvolver um protótipo capaz de testar a viabilidade da utilização de HTML para representação de imagens em páginas *web*, avaliando custo/tempo de download dessas imagens. Este protótipo converteu imagens existentes na *web* ou de arquivos, mostrando em diversos tipos de compactação possíveis que serão desenvolvidas.

Após a fase de protótipo foi desenvolvido o modelo de representação DIVMAGE em páginas que contenham imagens mantendo o direito autoral, o usuário vê a imagem, mas em um modelo capaz de manter sua qualidade dificultando a cópia, assim mantendo o direito da imagem ao seu autor.

Os resultados são mostrados através de testes em sites, reproduzindo imagens que o desenvolvedor definir nos filtros do *Tomcat*, atendendo a viabilidade dos internautas e satisfazendo o proprietário das imagens.

### 4. Estado da arte

Esta seção trata dos principais assuntos relacionados a execução deste trabalho apresentando as terminologias que serão adotadas para apresentar a solução proposta.

#### 4.1 Computação gráfica

A Computação Gráfica (CG) é uma área da Ciência da Computação que se dedica ao estudo e desenvolvimento de técnicas e algoritmos para a geração (síntese) de imagens através do computador. Atualmente, a CG está presente em quase todas as áreas do conhecimento humano, desde o projeto de um casa até o desenvolvimento de ferramentas de entretenimento, entre as quais os jogos eletrônicos. (MANSSOUR, 2006)

Segundo ET Santos (2001), a Computação Gráfica é a área da Ciência da Computação responsável pela criação de “imagens bi e tridimensionais usadas para a pesquisa científica, para fins artísticos e, nas indústrias, para projetar, testar e anunciar produtos.” As Interfaces Gráficas de Usuário (GUI) e os sistemas multimídia tais como a World Wide Web, permitem aos usuários de computador selecionar imagens para executar ordens, eliminando a necessidade de memorizar comandos.

A computação gráfica atualmente é uma área que engloba, para melhor descrição didática, pelo menos três grandes subáreas: a Síntese de Imagens, o Processamento de Imagens e a Análise de Imagens. A figura 1 representa estas áreas.

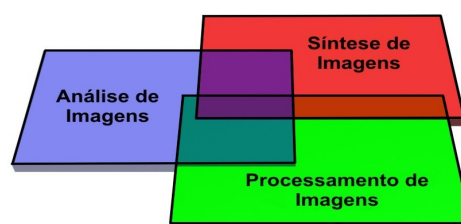


Figura 1. Grandes áreas da Computação Gráfica.

### 4.1.1 Síntese de Imagens

A Síntese de Imagens considera a criação sintética das imagens, ou seja, as representações visuais de objetos criados pelo computador a partir das especificações geométricas e visuais de seus componentes. Pode também ser descrita como Visualização Científica ou Computacional, principalmente quando se preocupa com a representação gráfica da informação, de forma a facilitar o entendimento de conjuntos de dados de alta complexidade, como, por exemplo, os dados de dinâmica dos fluidos, ou simulações espaciais.

### 4.1.2 Análise de Imagens

Esta área tem por finalidade obter, a partir de uma imagem (entrada), informações geométricas, parâmetros descritivos (em geral numéricos) que representam de modo sucinto informações importantes da imagem. As técnicas desta área tem grande interesse na área da robótica, com o objetivo de prover o sentido da visão aos robôs. Também usada para a realização de Reconhecimento de Padrões, Visão Computacional ou de extração de conhecimento das imagens (Mineração de Imagens)(GOMES,1998).

### 4.1.3 Processamento de Imagens

O Processamento de Imagens considera o processamento das imagens na forma digital e suas transformações, por exemplo, para melhorar ou realçar suas características visuais.

## 4.2 Sistema de processamento de imagens e seus componentes

Os elementos de um sistema de processamento de imagens de uso genérico são mostrados na figura 2. Este diagrama permite representar desde sistemas de baixo custo até sofisticadas estações de trabalho utilizadas em aplicações que envolvem intenso uso de imagens. Ele abrange as principais operações que se pode efetuar sobre uma imagem, a saber: aquisição, armazenamento, processamento e exibição. Além disso, uma imagem pode ser transmitida à distância utilizando meios de comunicação disponíveis. Todas estas operações são descritas a seguir.

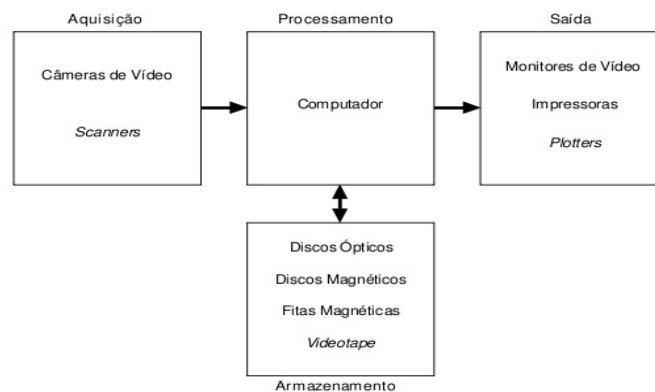


Figura 2. Elementos de um sistema de processamento de imagens.

### 4.2.1 Aquisição

A etapa de aquisição tem como função converter uma imagem em uma representação numérica adequada para o processamento digital subsequente. Este bloco compreende dois elementos principais. O primeiro é um dispositivo físico sensível a uma faixa de energia no espectro eletromagnético (como raio X, ultravioleta, espectro visível ou raios infravermelhos), que produz na saída um sinal elétrico proporcional ao nível de energia detectado. O segundo o digitalizador propriamente dito converte o sinal elétrico analógico em informação digital, isto é, que pode ser

representada através de *bits* 0s e 1s. Um módulo de aquisição de imagens é normalmente conhecido pelo nome de *frame grabber*.(MARQUES, 1999)

#### 4.2.2 Armazenamento

O armazenamento de imagens digitais é um dos maiores desafios no projeto de sistemas de processamento de imagens, em razão da grande quantidade de *bytes* necessários para tanto. Este armazenamento pode ser dividido em três categorias:

1. Armazenamento de curta duração de uma imagem: Quando o armazenamento é somente para o processamento da imagem, neste caso utiliza-se a memória RAM para se obter um processamento mais rápido.
2. Armazenamento para operações de recuperação de imagens relativamente rápidas: Quando há a necessidade de armazenar várias imagens para um processamento, pois a memória RAM não suportaria todas estas imagens.
3. Arquivamento de imagens: Quando necessita arquivar uma imagem no disco rígido (HD).

#### 4.2.3 Processamento

O processamento de imagens digitais envolve procedimentos normalmente expressos sob forma algorítmica. Em função disto, com exceção das etapas de aquisição e exibição, a maioria das funções de processamento de imagens pode ser implementada via software. O uso de hardware especializado para processamento de imagens somente é necessário em situações nas quais certas limitações do computador principal (por exemplo, velocidade de transferência dos dados através do barramento) forem intoleráveis.

A tendência atual do mercado de hardware para processamento de imagens é a comercialização de placas genéricas compatíveis com os padrões de barramento consagrados pelas arquiteturas mais populares de microcomputadores e estações de trabalho. O software de controle destas placas é que determinará sua aplicação específica a cada situação. As vantagens mais imediatas são: redução de custo, modularidade, reutilização de componentes de software em outra aplicação rodando sobre o mesmo hardware e independência de fornecedor. Convém notar, entretanto, que sistemas dedicados continuam sendo produzidos e comercializados para atender a tarefas específicas, tais como processamento de imagens transmitidas por satélites.

#### 4.2.4 Saída

A saída é a exibição da imagem após todo seu processo, podendo ser visualizada em várias formas, como monitores de vídeo, *plotters*, impressoras, etc..

### 4.3 Etapas do Processamento de Imagem

As etapas do processamento de imagem são divididas conforme a figura 3.

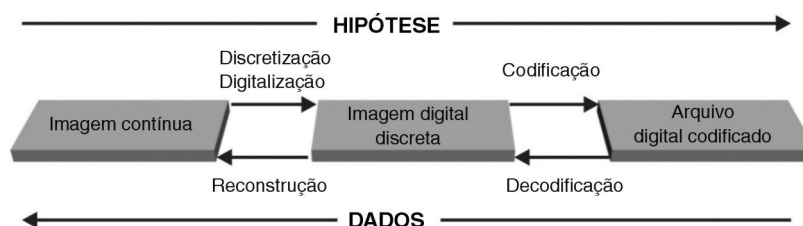


Figura 3. Etapas do processamento de imagens

- Discretização: conversão da imagem na forma contínua em um uma representação discreta.
- Reconstrução: processo inverso da discretização.
- Codificação: a partir da representação discreta da imagem, gera um conjunto de dados representativos da imagem, dados estes que podem ser transformados no formato de arquivos.
- Decodificação: processo oposto à codificação no qual acessam-se informações codificadas na forma de uma representação discreta.

#### 4.4 Java2D

Quando utilizamos a linguagem java para programar o processamento de imagem utilizamos a API Java 2D. A API Java 2D é um conjunto de classes de avançados gráficos 2D e de imagem, que engloba arte de linha, texto e imagens em um modelo único e abrangente. A API fornece amplo suporte para a imagem de composição e imagens *Alpha Channel*, um conjunto de classes para assegurar uma definição precisa do espaço de cor e conversão, e um rico conjunto de operadores de exibição de imagem orientada. (SUN)

#### 4.5 Tomcat

O Tomcat é um servidor de aplicações Java para web. É software livre e de código aberto desenvolvido dentro do conceituado projeto Apache Jakarta e oficialmente endossado pela Sun como a Implementação de Referência (RI) para as tecnologias Java Servlet e JavaServer Pages (JSP). O Tomcat é robusto e eficiente o suficiente para ser utilizado mesmo em um ambiente de produção. Tecnicamente, o Tomcat é um Container Web, parte da plataforma corporativa Java Enterprise Edition (J2EE ou Java EE) que abrange as tecnologias Servlet e JSP, incluindo tecnologias de apoio relacionadas como Realms e segurança, JNDI Resources e JDBC DataSources.

O Tomcat tem a capacidade de atuar também como servidor web/HTTP, ou pode funcionar integrado a um servidor web dedicado como o Apache httpd ou o Microsoft IIS.

#### 4.7 Filtro

*Servlet Filters* (Filtro) é um recurso do Java EE para interceptar uma requisição web antes do recurso final ser atingido, este recurso pode ser um *servlet*, um arquivo, entre outros. Filtros são muito úteis quando precisamos executar uma funcionalidade antes de que o *servlet* seja acessado, como por exemplo, verificar se um usuário tem a permissão para acessar a página requisitada. Na figura 4 mostra um exemplo de um filtro.

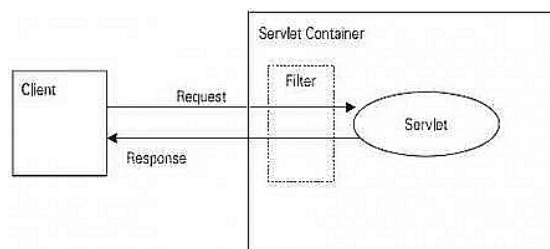


Figura 4. Exemplo de um Filtro

O filtro é uma interface que você **precisa implementar** ao escrever um filtro. O ciclo de vida de um filtro é representado por três métodos da interface *Filter*, o **init**, **doFilter** e **destroy**.

Init – Método que é acessado quando o filtro acaba de ser instanciado.

DoFilter – Considerado o coração do filtro, onde é executada toda a função que caberá ao filtro.

Destroy – Método executado após todo o processo do filtro.

## 4.8 HTML

A linguagem HTML (*HyperText Markup Language*) é um conjunto definido de *tags*, ou um DTD (*Document Type Definition*) específico do SGML (*Standard Generalized Markup Language*), e foi criada tendo em mente a necessidade de construção de documentos para serem exibidos em dispositivos de computador, daí sua vocação para tratar do formato que os dados contidos no documento vão assumir ao serem exibidos. Um navegador ou *browser*, ao ler um documento HTML, interpreta as *tags* que este documento contém para decidir como serão exibidos os dados também contidos. Os navegadores atuais interpretam o HTML porque o DTD para definição do HTML é fixo, e é conhecido *a priori* pelo interpretador do navegador. Assim mesmo, podemos ter navegadores diferentes interpretando definições de exibição de forma particular, com resultados distintos no dispositivo de saída. A estrutura do HTML é rígida, não existindo a possibilidade de adição de novos comandos de marcação (*tags*), sem que haja uma redefinição do DTD da linguagem e consequente atualização dos navegadores para que interpretem estas novas *tags*. A última especificação do HTML lançada pelo W3C foi a versão 4.0, e desde então a linguagem não tem sofrido mais modificações (SOUZA, 2004).

Entre as *tags* HTML a tag `<img>` é responsável pela renderização de imagens em uma página e a tag `<div>` define uma divisão ou uma seção em um documento HTML e é frequentemente utilizado para bloquear o grupo de elementos para formatá-los com estilos (W3SCHOOLS,2010). Este trabalho utilizará a tag `<div>` para a representação de imagens no documento HTML.

## 5 Desenvolvimento

Esta seção irá tratar o desenvolvimento do componente proposto.

### 5.1 Visão geral da ferramenta

O DIVMAGE é um componente capaz de analisar toda a página HTML requisitada, encontrar todas as imagens contidas na página e transforma-las em uma representação que evita cópias indevidas e sua reprodução. O componente DIVMAGE possui um mecanismo de filtro na aplicação, onde toda página requisitada passa por este filtro de análise e conversão das imagens.

### 5.2 Funcionalidade do componente DIVMAGE

Toda aplicação possui um diretório específico onde estão localizadas as imagens utilizadas, o componente exige a criação de uma pasta específica das imagens que necessitam evitar plágio, e esta pasta é definida na configuração do filtro do componente. O cliente solicita uma página, esta página passa pelo filtro que irá localizar as imagens que estão especificada na pasta de configuração do filtro, caso haja uma ou mais imagem que necessite a conversão o filtro transforma a imagem na representação DIVMAGE e responde ao cliente esta representação, juntamente com a função que será executada no cliente capaz de exibir a imagem a partir do modelo.

A funcionalidade do componente é definida em 6 passos conforme a figura 5.

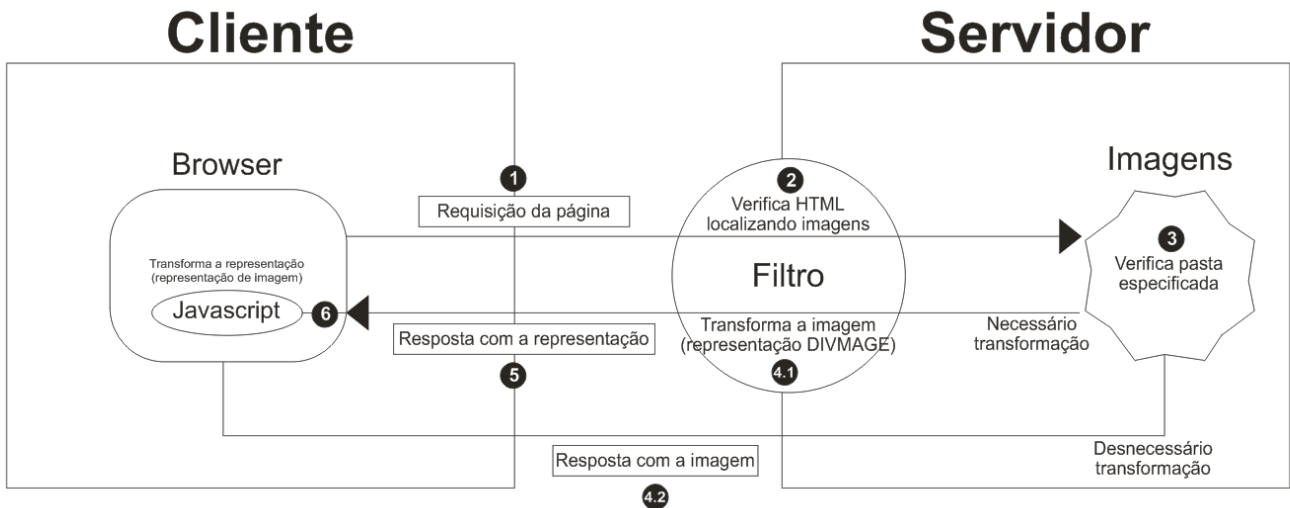


Figura 5. Funcionalidade do componente DIVMAGE.

1. O cliente requisita uma página;
2. A página solicitada passa pelo filtro, que analisa todo o conteúdo HTML, localizando imagens.
3. Ao encontrar uma imagem o filtro busca esta imagem na pasta especificada e analisa se a imagem encontrada será transformada ou não.
4.
  1. Caso seja necessário a transformação, a imagem é transformada no modelo de representação DIVMAGE.
  2. Caso seja desnecessário a transformação a imagem é retornada ao *browser*.
5. As imagens após transformadas são encaminhadas para o cliente juntamente com uma função *Javascript*.
6. Esta função *Javascript* é aberta pelo *browser* e transforma o modelo de representação DIVMAGE em uma representação de imagem.

Para representação desta imagem há dois processamentos: um do lado do cliente e outro do servidor. O processamento do lado do servidor é executado por um filtro que transforma a imagem original em uma representação desta imagem. Ao processamento do lado cliente cabe transformar esta representação em tags HTML de maneira que a visualização da imagem é garantida sem que tenha existido a transferência do arquivo de imagem.

### 5.3 Filtro do componente DIVMAGE

Conforme apresentado na seção XX, um filtro de aplicação é capaz de alterar o conteúdo de todas as requisições e respostas para uma determinada URL. Este filtro é configurado no arquivo `web.xml` da aplicação como mostra a figura 6.

```

<filter>
  <display-name>FiltroImagem</display-name>
  <filter-name>FiltroImagem</filter-name>
  <filter-class>br.uem.FiltroImagem</filter-class>
  <init-param>
    <param-name>dir</param-name>
    <param-value>/img</param-value>
  </init-param>
  <init-param>
    <param-name>comp</param-name>
    <param-value>>false</param-value>
  </init-param>
</filter>
<filter-mapping>
  <filter-name>FiltroImagem</filter-name>
  <url-pattern>*/url-pattern</url-pattern>
</filter-mapping>

```

Figura 6. Configuração do Filtro no arquivo `web.xml` da aplicação

Para a configuração do filtro é necessário conhecer as tags principais:

- <filter> - Configuração do filtro
- <display-name> - Nome utilizado na visualização do filtro
- <filter-name> - Nome do filtro
- <filter-class> - Caminho absoluto da classe que implementa o filtro
- <init> - Parâmetros de configuração do filtro
- <filter-mapping> - Mapeamento do filtro
- <filter-name> - Nome do filtro (mesmo nome utilizado na configuração do filtro)
- <url-pattern> - Define qual requisição seja filtrada. (\* referência a todas as requisições)

Este filtro possui dois parâmetros de configuração para sua execução:

**dir** – Parâmetro que define o diretório onde está contida as imagens que serão convertidas. Este parâmetro é obrigatório para a execução do filtro;

**comp** – Parâmetro que define se a conversão possuirá compressão utilizando “true” ou não utilizando “false”. Este parâmetro não é obrigatório para a execução, seu valor padrão é “false”

### 5.4 Modelo de representação do componente DIVMAGE

As imagens digitais são representadas através de *pixels*. O pixel é uma amostra da imagem “quantizada” em valores inteiros. Todo *pixel* possui uma posição e uma cor específica. A figura 7 mostra um exemplo de uma imagem e ao lado os *pixels* correspondente a amostra da imagem no padrão de cor RGB.

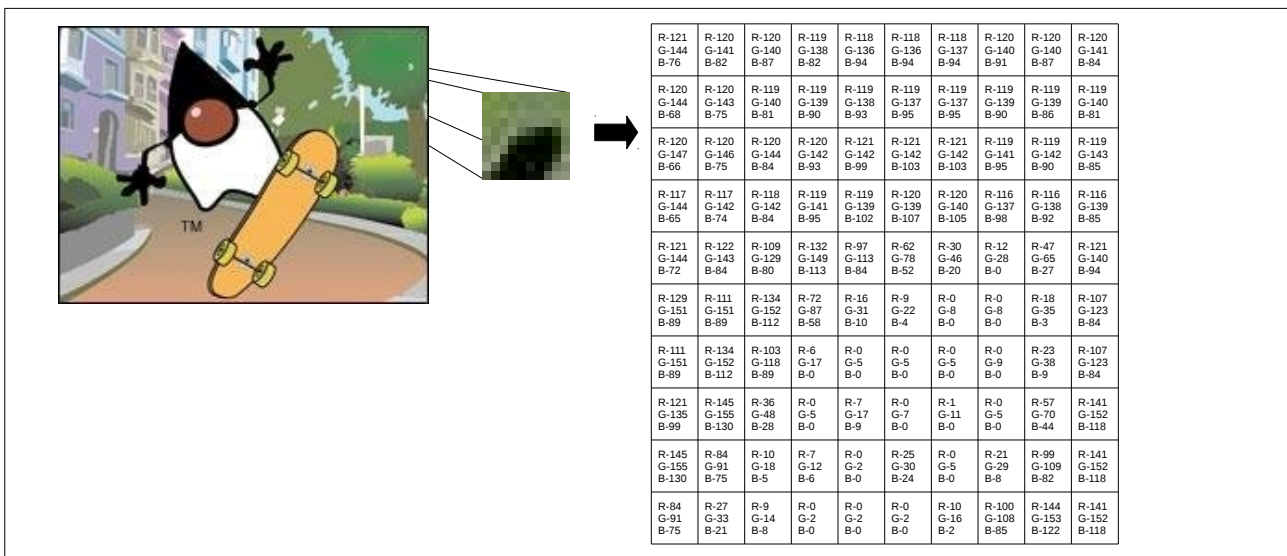


Figura 7. Representação Matricial com tons RGB.

A representação do DIVMAGE na parte do servidor faz com que a imagem contida na página se transforme em um elemento <div> que contem pixel a pixel da imagem transformada. A figura 8 representa o modelo do componente referente a amostra da imagem na figura 7.

```
<div class='imagem' style='width:10px;height:10px; background-color:red;'>#1;728c42#1;7f9952#1;70884a#1;7b925c#2;728859#1;788d62#1;73895b#1;768d59#1;738a54#0;738a54#1;7b954c#1;728b47#1;758d51#1;6b824c#1;788e60#1;70855c#1;71865d#1;7c9264#1;748a59#1;6f8650#1;768d49#1;7c9252#1;859a61#1;788c5b#1;82946c#1;7b8d67#1;798b65#1;7a8c64#1;819466#1;7e925f#0;7e925f#2;778d4f#1;6a7e4b#1;7d9063#1;748561#1;778866#1;7b8c6a#1;758662#1;6a7c52#1;7f9264#0;7f9264#1;859861#1;6e804e#1;798a5e#1;899974#1;606f50#1;435137#1;1e2c12#1;000000#1;34441f#1;73845a#0;73845a#1;7c8c5f#1;6c7d51#1;85936f#1;404f30#1;142008#4;000000#1;687853#0;687853#1;727e58#1;889672#1;727d5f#5;000000#1;1b250a#1;717f5e#0;717f5e#1;7d8868#1;949f81#1;212b13#5;000000#1;3a442b#1;919c7e#0;919c7e#1;939b83#1;535b44#3;000000#1;1c211b#1;000000#1;1b2113#1;6a725b#1;7d866b#0;7d866b#1;555d46#1;202813#5;000000#1;5e6456#1;868e79#1;8d957d#0;8d957d</div>
```

Figura 8. Modelo de representação DIVMAGE

Esta representação é processada no servidor de aplicação da página e retornada ao cliente. Anexada a esta representação existe uma função *Javascript* que é responsável pela representação da imagem a partir do modelo DIVMAGE.

## 5.5 Javascript

O processamento do componente DIVMAGE poderia ser todo no lado do servidor, porém o custo de transferência da imagem renderizada DIVMAGE poderia inviabilizar o acesso a essas imagens em rede com acesso mais limitado em relação a conexão. Desta maneira a opção deste projeto foi dividir o processamento entre cliente e servidor, sendo que o cliente fica responsável por representar a imagem. Como esse processamento no cliente vai ser feito via *browser* optamos utilizar a linguagem *Javascript*.

Javascript é uma linguagem de programação *client-side*, ou seja, seu processamento é no *browser* do cliente. A representação DIVMAGE demonstrada na figura 8 é transmitida para o cliente juntamente com uma função Javascript. Esta função Javascript é capaz de, a partir do modelo de representação, renderizar pixel a pixel a imagem.

## 6. Resultados obtidos

As conversões efetuadas pelo componente DIVMAGE garante a apresentação de uma imagem sem que o mesmo faça *download* desta imagem. Assim os usuário não conseguiram salvar a imagem através do atalho do navegador que permite clicar com o botão direito do mouse e salvar a imagem específica. A opção “salvar como...” da imagem processada não salva a imagem, salva a página HTML que está ativa no navegador.

A Figura 9 mostra a diferença do clique com o botão direito da imagem original e da imagem processada pelo componente aqui apresentado.

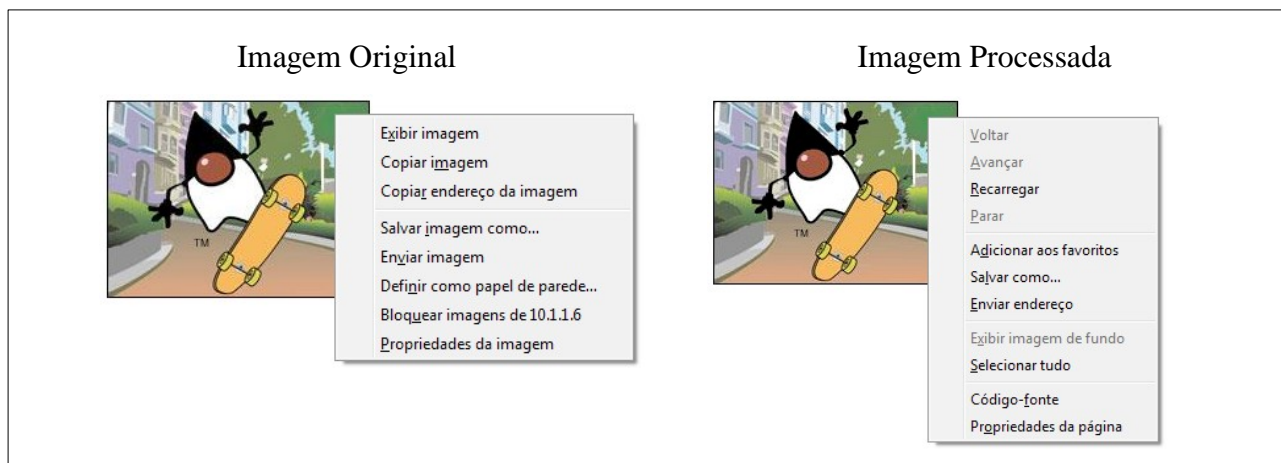


Figura 9. Diferença do clique com o botão direito no navegador

Além disto, como a imagem não foi enviada para o cliente a mesma não estará no *cache* do navegador.

## 7. Conclusão

No começo da era da Internet era difícil ver uma imagem que possui-se uma resolução adequada de visualização devido as limitações de velocidade, no começo era mais utilizados GIFS, e imagens com resoluções baixas. Atualmente a Internet se popularizou muito e ganhou muito mais

infraestrutura para suportar diversos tipos e tamanhos de imagens. Ao disponibilizar uma imagem em um site qualquer, esta imagem poderá ser vista e copiada de qualquer lugar do mundo, ainda mais que hoje em dia existe muitos buscadores de imagens, que conseguem localizar qualquer tipo de imagem que deseja.

O DIVMAGE tem o intuito evitar o plágio em imagens cujo o proprietário gostaria de disponibilizar imagens divulgando seu trabalho, seu talento, ou até mesmo sua profissão sem que seja prejudicado, divulgando uma imagem sem que outra pessoa utilize para outros ideais. A qualidade da imagem será mantida, mas dependendo do tamanho da imagem a conversão pode ficar um pouco mais lenta em relação a imagem mostrada normalmente.

## 8. Trabalhos futuros

O DIVMAGE possui algumas limitações por somente possuir um tipo de compressão. Futuramente seria viável desenvolver outras formas de compressão deixando a disposição do utilizador escolher o algoritmo de compressão desejado, também analisar o método de conversão otimizando para ganhar um melhor desempenho.

O processamento é feita no servidor e no cliente, a parte do cliente está com o processamento maior. Será analisado o custo do processamento no lado do cliente, alterando parte deste processamento para o servidor diminuindo o processamento no lado do cliente ou será alterado a maneira do processamento do cliente sempre viabilizando o melhor desempenho na representação da imagem.

## Referências

- ABRÃO, Eliane Yachou. **Imagem, Fotografia e Direitos Autorais**: [http://www2.uol.com.br/direito\\_autoral/artigo02.htm](http://www2.uol.com.br/direito_autoral/artigo02.htm), Acesso em março 2009.
- ALEXANDER, Bryan. **Web 2.0: A new wave of innovation for teaching and learning?** EDUCAUSE Review, v. 41, n. 2, p. 32-44, 2006. Disponível em: <<http://www.educause.edu/ir/library/pdf/ERM0621.pdf>>. Acesso em março de 2010.
- Apache Tomcat. <http://tomcat.apache.org/>. Acesso em março 2010.
- ASCENSÃO, José de Oliveira. **Direito autoral**. Rio de Janeiro: Forense, 1980.
- AZEVEDO, Eduardo, CONCI, Aura. **Computação Gráfica Teoria e Prática**. 1 edição. 2003.
- Bittar, C. A. , **Direito de Autor**. Editora: Forense Universitária. 4ª edição – 2004.
- BRASIL. **Lei nº 9.610**, de 19 de fevereiro de 1998. Disponível em [http://www.planalto.gov.br/ccivil\\_03/leis/19610.htm](http://www.planalto.gov.br/ccivil_03/leis/19610.htm). Acesso em 21/03/2009
- BRASIL. **Lei nº 10.695**, de 01 de julho de 2003. Disponível em <http://www.planalto.gov.br/CCIVIL/Leis/2003/L10.695.htm>>. Acesso em 21/03/2009
- BRESSAN, R. T. , **YouTube: intervenções e ativismos**. Universidade Federal de Juiz de Fora (UFJF), 2007.
- CAMPANI, Carlos A. P.. **Avaliação da Compressão de Dados e da Qualidade de Imagem em Modelos de Animação Gráfica para Web: uma nova abordagem baseada em Complexidade de Kolmogorov**. PhD thesis, Universidade Federal do Rio Grande do Sul, 2005. 111p.
- CARVALHO, Adriano A., **Estudo e Implementação de Algoritmos Clássicos para Processamento Digital de Imagens**. Dezembro, 2003 . Monografia (Bacharel Ciência da

Computação) Universidade Federal de Lavras.

- CONCI, Aura; AZEVEDO, Eduardo ; LETA, Fabiana R.. **Computação gráfica [v.2]**. - Rio de Janeiro : Elsevier, 2008.
- DEGUCHI, Luis Gustavo Seki. **Violações dos direitos autorais na Internet**. Novembro , 2007. Monografia (Bacharel em Direito) - Faculdades Integradas “Antônio Eufrásio de Toledo” .
- Deitel, Harvey M. **Java como Programar**. 6º edição, São Paulo : Pearson, 2006.
- ET Santos, MK Zuffo, ML Netto, R de Deus Lopes - **Computação gráfica: estado da arte e a pesquisa na usp**, 2001 – lsi.usp.br
- FRANKLIN, Erick C. S., **Desenvolvimento de uma metodologia para captura de imagens usando microcontroladores**, dezembro, 2005. Trabalho de Conclusão de Curso Engenharia da Computação. Escola Politécnica de Pernambuco – Universidade de Pernambuco.
- GANDELMAN, Henrique. **De Gutenberg à Internet: direitos autorais na era digital**. 5 ed. Revista e atualizada. Rio de Janeiro: Record, 2007.
- GOMES, Jonas; VELHO, Luiz. **Computação Gráfica** , Rio de Janeiro: IMPA, 1998.
- AGOSTINI, i silva, s bampi - **projeto de arquitetura de codificador de entropia para compressão jpeg de imagens em tons de cinza** - workshop iberchip, 2002 - iberchip.org
- MANSO, Eduardo Vieira. **A informática e os direitos intelectuais**. São Paulo: RT, 2000.
- MANSSOUR, Isabel Harb; COHEN, Marcelo. **Introdução à Computação Gráfica**. Revista RITA, 2006, vol. 13, nº 2.
- MARQUES Filho, Ogê; VIEIRA NETO, Hugo. **Processamento Digital de Imagens**, Rio de Janeiro: Brasport, 1999.
- MARTINS, Plínio F. Ci. Inf., Brasília, v. 27, n. 2, p. 183-188, maio/ago. 1998
- MANTOVANI, Osmar ; DIAS, Maria H. P. ; LIESENBERG, Hans , **Conteúdos abertos e compartilhados: novas perspectivas para a educação**. Janeiro de 2006
- MELLO, Cláudio G. **O Padrão JPEG e o seu Uso na Transmissão de Imagens Adaptativas ao Retardo na Rede**. Instituto Militar de Engenharia.
- MONTEIRO, L. , **A Internet como meio de comunicação: Possibilidade e Limitações**. INTERCOM – Sociedade Brasileira de Estudos Interdisciplinares da Comunicação XXIV Congresso Brasileiro da Comunicação – Campo Grande /MS – setembro 2001
- NETO, Roberto Hartke. Curso de JSP. Universidade Federal de Santa Catarina. **Programa Especial de Treinamento Ciências da Computação**. Versão 1.0. Florianópolis, Outubro de 2002.
- O'REILLY, What is Web 2.0: <http://oreilly.com/web2/archive/what-is-web-20.html> . Acesso em março 2010.
- ORRICO, Hugo Jr., Pirataria de software. Ed. 2. São Paulo, 2004, pg -116.
- PECK, Patricia. **Direito Autoral na Era Digital**:[http://www.patriciapeck.com.br/conhecimento .asp?Passo=Exibir&Materia=18](http://www.patriciapeck.com.br/conhecimento.asp?Passo=Exibir&Materia=18) , acesso em março 2009.

- PEREIRA, A. M., PIMENTEL, L. O. , MEHLAN V. . **Direitos Autorais: estudos e considerações.** Florianópolis, 12 a 14 de novembro de 2003.
- PORTO, Mayla Yara. **A Multimídia, a Pirataria Musical e o Direito Autoral:** [http://www.iar.unicamp.br/disciplinas/am625\\_2003/Mayla\\_artigo.html](http://www.iar.unicamp.br/disciplinas/am625_2003/Mayla_artigo.html) , Acesso em março 2009.
- RAMOS, O. T. H. **Análise Comparativa entre os principais formatos de armazenamento de imagens.** Dissertação de Mestrado pela Universidade Presbiteriana Mackenzie. São Paulo, dezembro de 2000.
- SOUZA, Renato Rocha; ALVARENGA, Lídia. **A Web Semântica e suas contribuições para a ciência da informação.** Ci. Inf. [online]. 2004, vol.33, n.1, pp. 132-141.
- SUN, Java 2D API : <http://java.sun.com/products/java-media/2D/index.jsp>. Acesso em março 2010.
- TANENBAUM, A. S. , **Redes de Computadores** / Andrew S. Tenenbaum ; tradução Vendenberg D. de Souza – Rio de Janeiro : Elsevier, 2003 – 11 reimpressão.
- VIANNA, Arlindo Cardarett, **Computação Gráfica**, Tecgraf/PUC-Rio, Apostila (disponível em PDF), Ago. 1996.
- W3SCHOOLS, HTML <DIV> tag: [http://www.w3schools.com/TAGS/tag\\_div.asp](http://www.w3schools.com/TAGS/tag_div.asp) . Acesso em março 2010.