

**UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
CÂMPUS PONTA GROSSA
DEPARTAMENTO ACADEMICO DE INFORMÁTICA - DAINF
COORDENAÇÃO DO CURSO SUPERIOR DE TECNOLOGIA EM ANÁLISE E
DESENVOLVIMENTO DE SISTEMAS – COADS**

**CASSIANO JULIO
GUILHERME ORMELEZ BECHEL**

**IDENTIFICAÇÃO DE CORES PREDOMINANTES EM FOTOS DE
CÂMERAS DE SMARTPHONES ATRAVÉS DE CHAMADAS
ASSÍNCRONAS**

TRABALHO DE CONCLUSÃO DE CURSO

**PONTA GROSSA
2013**

**CASSIANO JULIO
GUILHERME ORMELEZ BECHEL**

**IDENTIFICAÇÃO DE CORES PREDOMINANTES EM FOTOS DE
CÂMERAS DE SMARTPHONES ATRAVÉS DE CHAMADAS
ASSÍNCRONAS**

Trabalho de Conclusão de Curso apresentada como requisito parcial à obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas do Departamento Acadêmico de Informática DAINF / Coordenação do Curso de Análise e Desenvolvimento de Sistemas - COADS, da Universidade Tecnológica Federal do Paraná.

Orientador: Prof. Thalita Scharr Rodrigues

**PONTA GROSSA
2013**



Ministério da Educação
**Universidade Tecnológica Federal do
Paraná**
Campus Ponta Grossa



Diretoria de Graduação e Educação
Profissional

TERMO DE APROVAÇÃO

IDENTIFICAÇÃO DE CORES PREDOMINANTES EM FOTOS DE CÂMERAS DE
SMARTPHONES ATRAVÉS DE CHAMADAS ASSÍNCRONAS

por

CASSIANO JULIO
GUILHERME ORMELEZ BECHEL

Este Trabalho de Conclusão de Curso (TCC) foi apresentado em 25 de março de 2013 como requisito parcial para a obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas. Os candidatos foram arguidos pela Banca Examinadora composta pelos professores abaixo assinados. Após deliberação, a Banca Examinadora considerou o trabalho aprovado.

Thalita Scharr Rodrigues
Profa. Orientadora

Wellton Costa de Oliveira
Membro titular

Robson Duda
Membro titular

Helyane Bronoski Borges
Responsável pelos Trabalhos
de Conclusão de Curso

Simone de Almeida
Coordenador do Curso
UTFPR - Campus Ponta Grossa

AGRADECIMENTOS

Cassiano Julio

Quero agradecer à Deus, que tornou possível mais essa realização na minha vida, dando forças nessa caminhada.

A todos meus amigos pelas risadas, apoio e companheirismo que me proporcionaram nessa passagem pela universidade.

Agradeço ao Guilherme, pelas parcerias nos trabalhos, projetos e nos feitos que realizamos.

Aos professores Thalita e Welton, pela motivação e pela ajuda no aumento de conhecimento, mostrando que ainda temos muito a trilhar.

Finalmente, agradeço aos meus pais, Pedro e Maria, pois me apóiam e incentivam em todos os meus sonhos.

Guilherme Ormelez Bechel

Aos meus avós Jacyra e Ovídio, por me darem todo apoio e estrutura para que pudesse concluir minha graduação.

A minha mãe Lucimar, que apesar de estar longe, jamais deixou de me apoiar em qualquer decisão.

A minha madrinha Edna, um obrigado mais do que especial, por tudo que você faz por mim e meus avós.

A minha namorada Denise, pela parceria e carinho de todos os dias e por aguentar todo o meu stress.

Ao meu colega Cassiano, e toda sua dedicação na realização deste trabalho.

A nossa orientadora Thalita, por todas suas palavras de motivação e seu comprometimento conosco.

A meus amigos, que de algum modo e intensidade participaram dessa jornada.

Aos meus colegas de turma, porque sem eles a faculdade não teria graça alguma.

“Quanto mais próximo o homem estiver de um desejo, mais o deseja; e se não consegue realizá-lo, maior dor sente.” Niccolò Maquiavel.

RESUMO

O presente trabalho tem como objetivo apresentar a utilização de fotografias capturadas por câmeras de *dispositivos móveis* utilizando o navegador e processamento de imagens. Para que seja realizado esse trabalho, serão utilizado as ferramentas HTML5, PHP, Browser Opera, MySql, Ajax . Assim demonstrando a utilização de uma câmera em um browser de *dispositivos móveis*, e as cores predominantes das imagens capturadas pela a câmera.

Palavras-chaves: Imagens. HTML5. OpenCV. Smartphone.

ABSTRACT

This paper aims to present the use of photographs captured by cameras from mobile devices using the browser and image processing. To accomplish this work, the tools will be used HTML5, PHP, Opera Browser, MySQL, Ajax. Thus demonstrating the use of a camera in a mobile browser, and the predominant colors of the images captured by the camera.

Keywords: images. html5. opencv. smartphone.

LISTA DE FIGURAS

Figura 1 - Círculos Cromáticos RGB.	14
Figura 2 - Valores das cores dentro do Espectro Visível ao olho Humano.....	15
Figura 3 - Percepção da cor pelo Comprimento da Onda.....	16
Figura 4 - Detalhe da primeira página das referências da composição das cores utilizadas por Fonte: Eiseman (2000,p.130-173).....	17
Figura 5 - Exemplo de Pixel.....	18
Figura 6 - Forma de captura de imagem por radiação.	19
Figura 7 - Etapas do processamento da imagem.	20
Figura 8 - Fórmula para cálculo dos elementos de um histograma.....	22
Figura 9 - Exemplo de Histograma.	22
Figura 10 - Exemplo de histograma para imagem com oito níveis de cinza.....	23
Figura 11 - Exemplo de escala de Intensidade.	23
Figura 12 - Intensidade e Frequência da Figura 11.	24
Figura 13 - Imagem colorida e imagens com as bandas RGB.....	24
Figura 14 - Histograma da imagem da Figura 13, separadas nas bandas do sistema de cores RGB.	25
Figura 15 - Código fonte da utilização da tag<canvas>.	27
Figura 16 - Resultado da exemplificação tag<canvas>.....	27
Figura 17 - Código Fonte Utilizando a tag <Video>.	28
Figura 18 - JavaScript para execução da tag<vídeo>.	28
Figura 19 - Imagem gerada com a tag<vídeo>.	29
Figura 20 - Código Fonte Utilizando a tag <Audio>.....	29
Figura 21 - Reprodutor de audio gerada com a tag<Audio>.	30
Figura 22 - Utilização do JavaScript.	31
Figura 23 - Modelo de aplicação Ajax.....	32
Figura 24 - Implementação usando Ajax.	33
Figura 25 - Chamada da biblioteca jquery.....	34
Figura 26 - Utilização de jquery.....	34
Figura 27 - Código Cliente HTML5 para chamada da Câmera do smartphone. .	37
Figura 28 - JavaScript trabalhando com o elemento vídeo do HTML5.....	38
Figura 29 - Experimento 1.....	40
Figura 30 - Experimento 2.....	41
Figura 31 - Experimento 3.....	42
Figura 32 - Experimento 4.....	43
Figura 33 - Experimento 5.....	44
Figura 34 - Experimento 6.....	45
Figura 35 - Experimento 7.....	46

LISTA DE SIGLAS

AJAX - Asynchronous Javascript And Xml

API - Application Programming Interface

CSS - Cascading Style Sheets

JSON - Javascript Object Notation

HTML - HyperText Markup Language

HTTP - HyperText Transfer Protocol

OPENCV - Open Source Computer Vision Library

XML - eXtensible Markup Language

SUMÁRIO

1 – INTRODUÇÃO	11
1.1 - CONTEXTUALIZAÇÃO.....	11
1.2 – OBJETIVO.....	12
1.3 – PROBLEMAS.....	12
1.4 – ESTRUTURA DO TRABALHO	12
2 - FUNDAMENTAÇÃO TEÓRICA	12
2.1 – PROCESSAMENTOS DE IMAGENS.....	12
2.1.1 – Cor	13
2.1.2 – Processos De Formação De Cores.....	14
2.1.3 – Sistema De Cores	15
2.1.4 – Pixel.....	17
2.1.5 – Contraste e Brilho.....	18
2.1.6- Imagens Digitais.....	18
2.1.7 – Etapas Do Processamento	20
2.1.8 – Tipos de Imagens	20
2.1.9 – Histograma.....	21
2.2 – TECNOLOGIAS WEB	25
2.2.1 – OPENCV.....	25
2.2.2 - SMARTPHONE	25
2.2.3 - HTML5	26
2.2.5 – JAVASCRIPT	30
2.2.4 - AJAX	31
2.2.6 – JQUERY.....	34
2.2.7 – FLASH LITE	35
2.2.8 – PHP	35
3 – METODOLOGIA	36
3.1 – CLIENTE – SMARTPHONE.....	36
3.2 – PROCESSAMENTO DE IMAGEM – PHP.....	38
4- RESULTADOS	40
4.1 - EXPERIMENTO 1	40
4.2 – EXPERIMENTO 2	41
4.3 – EXPERIMENTO 3	42
4.4 – EXPERIMENTO 4	43
4.5 – EXPERIMENTO 5	44
4.6 – EXPERIMENTO 6	45
4.7 – EXPERIMENTO 7	46

5 - CONCLUSÕES E PERSPECTIVAS DE PESQUISAS FUTURAS	48
6 - REFERENCIAS	49

1 – INTRODUÇÃO

Com a demanda de softwares na área de dispositivos móveis, existem algumas necessidades que não são atendidas. Pois os usuários utilizam os seus aparelhos móveis para realizar atividades do seu cotidiano, como verificar horários de ônibus, verificações cotações, realizar cotações.

Uma dessas necessidades é a utilização do *plugin Flashplayer* nos navegadores de smartphones. A maioria dos sites desenvolvidos tem menus criados com essa tecnologia, porém os celulares e os navegadores não dão suporte para essa tecnologia.

No presente trabalho visa-se apresentar a utilização do HTML5 como alternativa ao Flashplayer da Adobe. Outro assunto abordado é o processamento de imagens, as quais são úteis para uma gama de atividades que exige interpretação. Desse modo, devido a falta de detalhes percebidos pelo olho humano, se utilizada apenas a visão humana, decisões ou informações errôneas podem ser realizadas.

Sendo assim, o processamento digital de imagens utilizando o computador torna-se confiável, podendo cobrir todo o espectro da imagem e assim determinar todos os tipos de cores presentes. Para isso, pretende-se processar a imagem capturada pelo *Smartphone* e extrair a cor predominante, utilizando funções do PHP.

1.1 - CONTEXTUALIZAÇÃO

Com a utilização da câmera do smartphone, pretende-se capturar imagens de roupas que os usuários utilizam. Essa imagem é enviada para um servidor, posteriormente é processada e extrai-se a cor predominante. Feito isso, compara-se a mesma com registros de um banco de dados, o qual possui as cores mais utilizadas em previsões de signos astrológicos. Compara-se a cor da roupa do usuário com a cor dada pela previsão diária para o signo do mesmo, retornando se as mesmas são iguais ou distintas.

1.2 – OBJETIVO

O presente trabalho tem como objetivo principal identificar cores predominantes em fotos tiradas de câmeras de *smartphones*, e transmiti-las para um servidor web e classificar a roupa do usuário pela cor predominante na mesma, comparando a cor usada com a que o signo do usuário sugere para o dia.

1.3 – PROBLEMAS

Como o desenvolvimento está em progresso, os programadores terão muito trabalho para atualizar os sites criados para desktop, pois uma grande empresa que é a Adobe , relatou em seu site que não estarão mais progredindo em suas ferramentas para o desenvolvimento web para móbile. E desta maneira teria que se mudar todo o site para rodar algumas aplicações, e como ainda está sendo criada a linguagem de marcação HTML5, não temos todo o conhecimento para o desenvolvimento de um site conseguindo equiparar com a tecnologia flash e nem muitas fontes para aumentar o conhecimento e também os navegadores não estão suportando todas as tecnologias do HTML5.

1.4 – ESTRUTURA DO TRABALHO

Este trabalho está dividido em seis capítulos. Dessa maneira, o Capítulo 2 apresenta a fundamentação teórica, a qual contém conceitos de processamento de imagens e das tecnologias utilizadas para ambiente Web. Em sequência, será apresentada no Capítulo 3 a metodologia do trabalho.

O Capítulo 4 aborda os resultados da aplicação e o Capítulo 5, as conclusões referentes ao trabalho proposto e o ultimo capitulo que é o 6 estará mostrando as referencias do trabalho.

2 - FUNDAMENTAÇÃO TEÓRICA

2.1 – PROCESSAMENTOS DE IMAGENS

Sendo que a visão humana um dos sentidos mais avançados, e também um dos mais utilizados, não é surpresa que as imagens desempenham um grande papel

no cotidiano. Uma das motivações para o estudo de processamento de imagens se refere ao melhoramento da informação visual para que se tenha uma melhor interpretação sobre as questões, as demais motivações compreendem o armazenamento, transmissão e a representação destas informações.

2.1.1 – Cor

Segundo Scuri (2009), cor no ponto de vista físico refere-se à luz. A luz entra no espaço em forma de uma onda ou como uma partícula, verificando-se assim a natureza dual da luz. A partir do tamanho desta onda, consegue-se distinguir as cores.

As cores são produzidas por raios luminosos, que são absorvidos ou refletidos por objetos. Com o comprimento da onda refletida pelo objeto, consegue-se reconhecer as cores nos ambientes. Desse modo, em ambientes com pouca iluminação, consegue-se observar somente cores escuras, em ambientes com iluminação significativa, é possível que o humano visualize diversas cores.

Assim, para que os humanos consigam enxergar as demais cores, precisa-se que as cores primárias sejam “misturadas” para gerar as demais cores. Segundo Rocha (2010), para que enxergar a cor Branco, precisa-se que as cores primárias sejam combinadas de forma que se tenha 30% de Vermelho, 59% de Verde e 11% de Azul, sendo já a cor Preto somente existe quando à ausência de luz. A Figura 1 apresenta uma ilustração das combinações das cores primárias.



Figura 1 - Círculos Cromáticos RGB.

Fonte: Cor Luz, Cor Pigmento e os sistemas RGB e CMY – ROCHA, Carlos João.

Conseqüentemente, as demais cores apresentam quantidades das cores primárias diferentes, dessa maneira sendo possível gerar diversas outras cores. A seguir apresentam-se os processos de formação de cores.

2.1.2 – Processos De Formação De Cores

O processo de formação de cores é classificado em três grupos: aditivos, subtrativos e pigmentação.

- Processo Aditivo: neste processo vários raios de cores são somados e gera-se um raio de luz com comprimento diferente. Esse processo ocorre em monitores e televisões;

- Processo Subtrativo: acontece quando o raio de luz passa por algum objeto, o qual elimina alguns comprimentos de onda, resultando em uma nova cor. Um exemplo desse tipo de processo é a utilização de projetores para visualizar slides em um tela;

- Processo de Pigmentação: a luz interage com diversos eventos, sendo que o deles seria uma nova cor. Este processo é utilizado em impressões de documentos, como o de manufaturas;

2.1.3 – Sistema De Cores

A seguir são apresentadas informações sobre sistemas de cores utilizados atualmente.

- Sistema RGB: O modelo de cores RG foi criado com base na teoria de Percepção de Cores de Young-Helmholtz, que foi dada continuidade pelo cientista alemão Herman Von Helmholtz onde foi proposto que o olho continha apenas 3(três) tipos de receptores de cores. Esses receptores respondiam de maneira forte aos comprimentos de ondas das cores Vermelho(R), Verde(G) e Azul(B). Ainda, deduziu-se que cada receptor de cores nos olhos possui uma grande sensibilidade à incidências luminosas, de modo que o olho humano é sensível a radiação eletromagnética em uma faixa de 400 a 700 nanômetros. Ainda, na Figura 2 observa-se que as cores que compõem o sistema RGB estão dentro desta faixa.

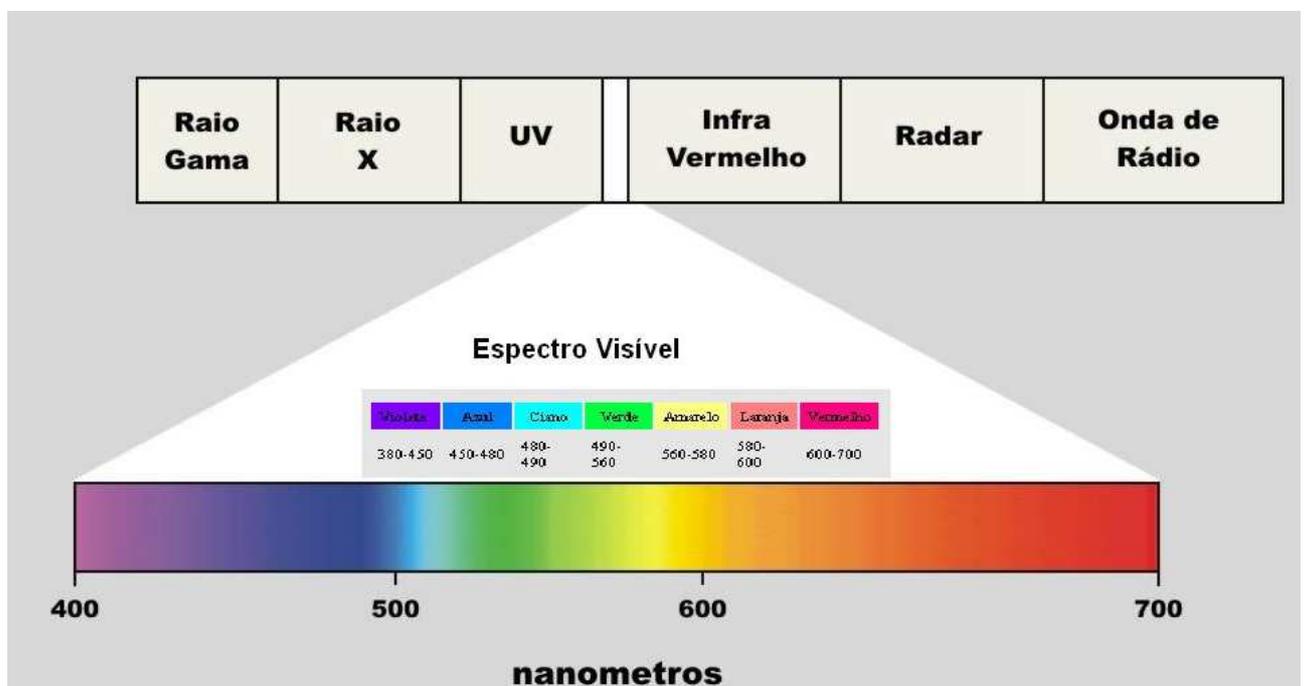


Figura 2 - Valores das cores dentro do Espectro Visível ao olho Humano.

Fonte: Fonte: Cor Luz, Cor Pigmento e os sistemas RGB e CMY – ROCHA, Carlos João.

A Figura 3 apresenta a faixa do comprimento de onda em que as cores podem ser compreendidas pelas pessoas. Como exemplo de utilização do sistema RGB têm-se os monitores de computadores, televisões e outros equipamentos.

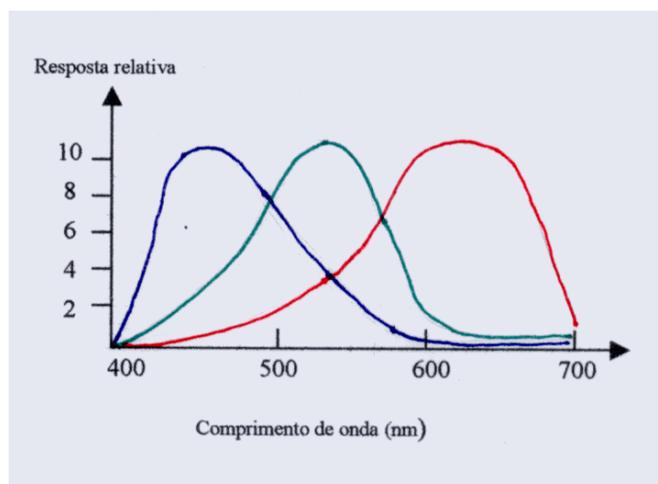


Figura 3 - Percepção da cor pelo Comprimento da Onda.

Fonte: <http://www.ic.uff.br/~aconci/curso/percep~1.htm>.

Para este sistema de cores também se utiliza o termo RGBA, onde o A significa *Alpha*. Não são modelos diferentes, porém uma nova representação, porque o A é utilizado para indicar o nível de transparência da cor. A utilização desta representação de cores representaria o efeito de turbidez, o qual é ocasionado pela atmosfera, tornando as cores com visualização opaca, em aproximação com a cor da realidade (SCURI, 1999).

– Sistema CMYK: O sistema de Cores CMYK, *Cyan*, *M - Magenta*, *Y - Yellow* e *K - Black*, também é chamado de cor-pigmento e é utilizado nas indústrias gráficas. O sistema CMYK é utilizado para impressões coloridas, podendo ocultar certas cores quando o fundo da impressão for da cor branca. Este tipo de impressão se chama subtrativo, pois quando se põe as cores com um fundo branco, o sistema de cores CMYK subtrai a luminosidade do branco. Com a combinação das cores que fazem parte do sistema, é possível gerar cores que estão dentro do espectro visível do ser humano, anteriormente apresentada na Figura 2.

-Sistema Pantone de Cores: Segundo Eiseman (2000), indica as cores utilizadas em suas combinações por meio da empresa referência Pantone. Ela é

uma empresa multinacional que desenvolve pigmentos especiais para usos diversos, desde tecido a papel. “Na produção gráfica, cores especiais, como as cores da escala Pantone, são aquelas utilizadas na impressão que não estão na quadricromia” (CMYK) (Henriques, 2004). Por causa da padronização de pigmentos, o resultado da impressão com o sistema Pantone será o mesmo, independente do país. A Figura 4 apresenta as cores utilizadas por esse sistema.

Color Conversion
Ink Color Formulas for Printing

PANTONE Color	PANTONE Process Color	PANTONE Hexachrome Color	PANTONE Color	PANTONE Process Color	PANTONE Hexachrome Color
PANTONE Process Yellow	C-0 M-0 Y-100 K-0	K-0 C-0 G-0 M-0 Y-100 D-0	PANTONE 130	C-0 M-29 Y-11 K-0	K-0 C-0 G-0
PANTONE Process Magenta	C-0 M-100 Y-0 K-0	K-7 C-0 G-0 M-100 Y-72 D-10	PANTONE 135	C-0 M-41 Y-100 K-17	K-20 C-0 G-0
PANTONE Process Cyan	C-100 M-0 Y-0 K-0	K-0 C-75 G-5 M-0 Y-72 D-0	PANTONE 140	C-0 M-36 Y-100 K-23	K-21 C-0 G-0
PANTONE Process Black	C-0 M-0 Y-0 K-100	K-100 C-0 G-0 M-0 Y-0 D-0	PANTONE 145	C-0 M-19 Y-91 K-0	K-2 C-0 G-0
PANTONE Yellow	C-0 M-3 Y-100 K-0	K-0 C-0 G-0 M-0 Y-100 D-0	PANTONE 150	C-0 M-28 Y-76 K-0	K-4 C-0 G-0
PANTONE Yellow 512	C-0 M-18 Y-100 K-0	K-0 C-0 G-0 M-0 Y-100 D-0	PANTONE 155	C-0 M-15 Y-69 K-0	K-3 C-0 G-0
PANTONE 131	C-0 M-33 Y-100 K-0	K-0 C-0 G-0 M-0 Y-100 D-0	PANTONE 160	C-0 M-11 Y-62 K-0	K-0 C-0 G-0
PANTONE 136	C-0 M-39 Y-100 K-0	K-0 C-0 G-0 M-0 Y-100 D-0	PANTONE 165	C-0 M-7 Y-55 K-0	K-0 C-0 G-0
PANTONE 141	C-0 M-45 Y-100 K-0	K-0 C-0 G-0 M-0 Y-100 D-0	PANTONE 170	C-0 M-3 Y-48 K-0	K-0 C-0 G-0
PANTONE 146	C-0 M-51 Y-100 K-0	K-0 C-0 G-0 M-0 Y-100 D-0	PANTONE 175	C-0 M-1 Y-41 K-0	K-0 C-0 G-0
PANTONE 151	C-0 M-57 Y-100 K-0	K-0 C-0 G-0 M-0 Y-100 D-0	PANTONE 180	C-0 M-1 Y-34 K-0	K-0 C-0 G-0
PANTONE 156	C-0 M-63 Y-100 K-0	K-0 C-0 G-0 M-0 Y-100 D-0	PANTONE 185	C-0 M-1 Y-27 K-0	K-0 C-0 G-0
PANTONE 161	C-0 M-69 Y-100 K-0	K-0 C-0 G-0 M-0 Y-100 D-0	PANTONE 190	C-0 M-1 Y-20 K-0	K-0 C-0 G-0
PANTONE 166	C-0 M-75 Y-100 K-0	K-0 C-0 G-0 M-0 Y-100 D-0	PANTONE 195	C-0 M-1 Y-13 K-0	K-0 C-0 G-0
PANTONE 171	C-0 M-81 Y-100 K-0	K-0 C-0 G-0 M-0 Y-100 D-0	PANTONE 200	C-0 M-1 Y-6 K-0	K-0 C-0 G-0
PANTONE 176	C-0 M-87 Y-100 K-0	K-0 C-0 G-0 M-0 Y-100 D-0			

Figura 4 - Detalhe da primeira página das referências da composição das cores utilizadas por Fonte: Eiseman (2000,p.130-173).

2.1.4 – Pixel

O pixel é o elemento básico de uma imagem, sendo que o conjunto de pixels de uma imagem é organizado na forma de um tabuleiro de xadrez (SCURI, 1999), Essa organização, com simetria quadrada, se deve a facilidade de implementação eletrônica, tanto dos sistemas de aquisição quanto dos sistemas de visualização de imagens (ALBUQUERQUE, 2001). Dessa maneira, é possível definir que um pixel é normalmente quadrado.

A partir da notação de pixel, foi criada uma média para as qualidades das imagens, denominada de resolução. A resolução é medida pela quantidade de pixels em relação à altura e à largura da imagem. A Figura 5 apresenta uma ilustração do

elemento básico, pixel, e a matriz desses elementos que forma a imagem digital.

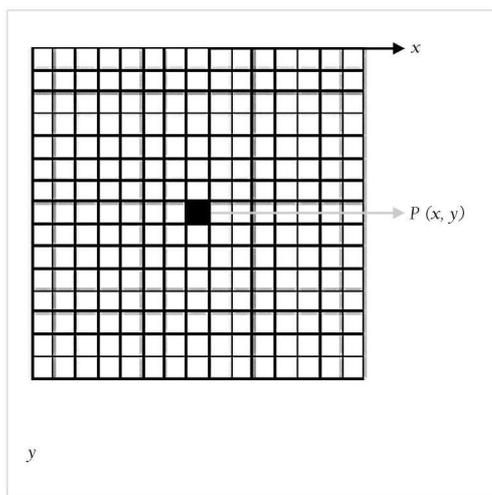


Figura 5 - Exemplo de Pixel.

Fonte: CONCI et al, 2008.

Desta maneira podemos dizer que uma câmera de 1.3 mega pixels é capaz de gerar uma imagem com 1.300.000(um milhão e trezentos mil) pixels.

2.1.5 – Contraste e Brilho

Existem duas características que são de grande importância para o processamento de imagens: o brilho e contraste. Quando analisa-se uma imagem, verifica-se que se a maioria dos pixels tem valores digitais perto do 0, a imagem é escura, assim a mesma não apresenta quantidade de brilho significativo. No entanto, quando os pixels têm valores altos, perto de 255, a imagem é considerada clara, assim verifica-se que a imagem tem brilho.

O contraste de uma imagem pode ser definido como o desvio padrão dos níveis digitais de todos os pixels da imagem (WEEKS, 1996).

2.1.6- Imagens Digitais

Conforme Marques Filho e Vieira Neto (1999), a palavra *imagem* vem do termo latim *imago*, que significa representação visual de um objeto qualquer. Na origem grega a palavra *imagem* corresponde ao termo *eidos*, que foi desenvolvido por Platão, que considerava a *Idea* (teoria do idealismo), da coisa a sua imagem

(uma projeção na mente).

Diferente de Platão, Aristóteles considerava a imagem como uma aquisição de sentidos, representando desse modo um objeto real, desta maneira foi criada a teoria do Realismo.

Imagens digitais são definidas por um plano limitado, onde cada posição nesse plano contém a informação de cor. Portanto, a imagem é um sinal de cor contínuo de duas dimensões, onde o domínio é o plano e o contradomínio é o espaço de cor (MARQUES FILHO e VIEIRA NETO, 1999).

A origem de uma imagem pode ser de ondas de radiação eletromagnéticas (Câmeras Fotográficas) ou também por ondas sonoras de alta frequência (Ultra som). Um exemplo de como as imagens são criadas, por meio de radiação pode ser visualizado na Figura 6.

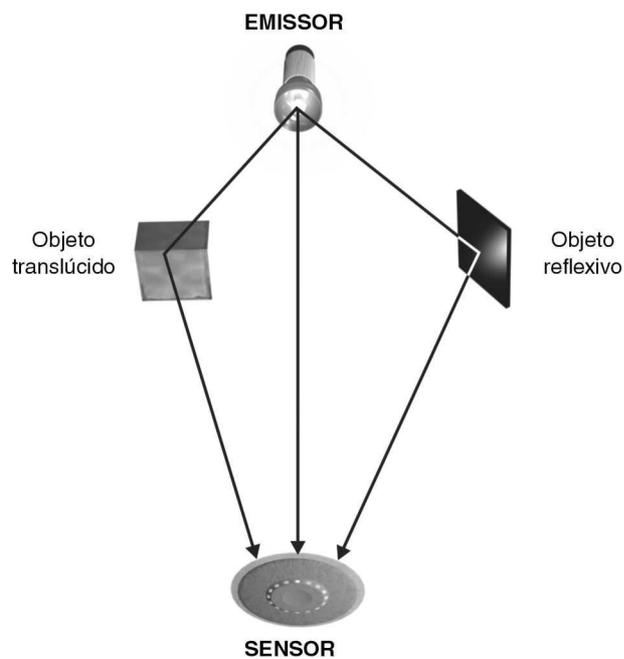


Figura 6 - Forma de captura de imagem por radiação.

Fonte: CONCI et al, 2008.

Para que uma imagem possa ser processada por computador, a mesma deve ser convertida para formato digital. Desse modo, a imagem poderá ser utilizada por sistemas e softwares computacionais, sendo possível também armazená-la na forma de arquivo, conforme pode ser visualizado na Figura 7.

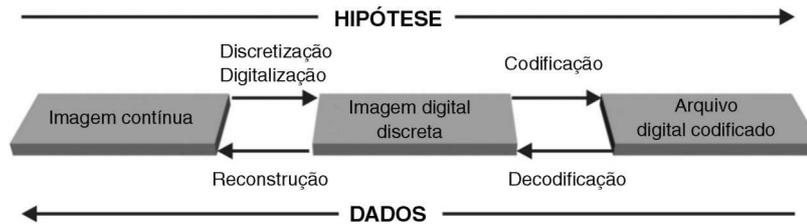


Figura 7 - Etapas do processamento da imagem.

Fonte: CONCI et al, 2008.

2.1.7 – Etapas Do Processamento

Para que a imagem seja interpretada e processada por computador, deverá passar pelas seguintes etapas:

- Discretização: Esta etapa envolve a conversão de uma imagem de forma contínua para uma forma discreta;

- Reconstrução: Este processo envolver a transformação para o formato contínuo, sendo que muitas vezes não é possível recuperar a imagem original;

- Codificação: Nesta etapa são modificadas as características de uma imagem, tornando-a apropriada para usos específicos. Um exemplo seria o armazenamento de uma imagem em algum formato, como Gif ou Jpeg, para que seja possível sua transmissão ou armazenagem;

- Decodificação: Como o próprio nome sugere, esse processo é oposto à codificação.

2.1.8 – Tipos de Imagens

A Imagem digital é uma imagem bidimensional, que emprega um código binário de modo a permitir o seu processamento, transferência, impressão ou reprodução. Existem dois tipos de imagens: *raster* e imagens vetoriais.

- Imagens de Rastreio (*raster*): É a representação em duas dimensões, como sendo um conjunto finito de pontos. Cada ponto é definido por valores numéricos formando uma malha, onde cada ponto é um pixel. Esse tipo de imagem tem uma limitação. Quando as dimensões são aumentadas, os pixels vão ocupar mais espaço, ou seja, essa malha vai se “esticar”, perdendo definições. Além disso, pode-

se considerar que a quantidade de pixel por polegada define a resolução de uma imagem, juntamente com o número de linhas e colunas da matriz de pixels existentes na imagem.

-Imagens Vetoriais: são geradas a partir de descrições geométricas de formas. Este tipo de imagem é composta por curvas, elipses, polígonos e outros elementos, ou seja, esse tipo de imagem é formada por vetores.

2.1.9 – Histograma

“O histograma de uma imagem é simplesmente um conjunto de números indicando o percentual de pixels naquela imagem que apresentam um determinado nível de cinza” (MARQUES FILHO E VIEIRA NETO, 1999, p.55). “Um histograma é, portanto uma representação gráfica da frequência de ocorrência de cada intensidade (nível de cor) de uma imagem” (CONCI et al, 2008).

Um histograma é construído por meio da análise do tom de cada pixel; fazendo a contagem do número de pixel de cada valor de intensidade; representando valores na forma de uma tabela ou em um gráfico da frequência correspondente a cada nível digital de intensidade, e por fim, se normaliza os valores dividindo pelo número total de pixels da imagem, obtendo uma frequência de cada tom.

A representação do histograma é realizada na forma de um gráfico de barras, sendo que cada barra apresenta um número ou percentagem de uma imagem em sua escala de cinza. Quando é visualizado um histograma, é possível indicar o contraste e o brilho da imagem, de acordo com a distribuição das colunas ao longo do histograma. A fórmula para o cálculo do histograma é apresentada na Figura 8. Exemplo (MARQUES FILHO e VIEIRA NETO, 1999,p.55) :

$$p_r(r_k) = \frac{n_k}{n}$$

$$0 \leq r_k \leq 1$$

$k = 0, 1, \dots, L-1$, onde L é o número de níveis de cinza da imagem digitalizada;

n = número total de pixels na imagem;

$p_r(r_k)$ = probabilidade do k -ésimo nível de cinza;

n_k = número de pixels cujo nível de cinza corresponde a k .

Figura 8 - Fórmula para cálculo dos elementos de um histograma.

Fonte: Marques Filho e Vieira Neto, 1999.

Os dados da Figura 9 correspondem a uma imagem de 128 x 128 pixels, com 8 níveis de cinza. O número de pixels correspondentes a um certo tom de cinza está indicado na segunda coluna, enquanto as respectivas probabilidades $p_r(r_k)$ aparecem na terceira coluna. A representação gráfica equivalente deste histograma é mostrada na Figura 10.

Conforme os mesmos autores, um histograma apresenta várias características importantes. A primeira delas é que cada $p_r(r_k)$ fornece, como sugere a notação, a probabilidade de um pixel da imagem apresentar nível de cinza r_k . Portanto, um histograma nada mais é que uma função de distribuição de probabilidades e como tal deve obedecer aos axiomas e teoremas da teoria de probabilidade. Por exemplo, é possível verificar que na Figura 9 a soma dos valores de $p_r(r_k)$ é 1, o que já era esperado.

Nível de cinza (r_k)	n_k	$p_r(r_k)$
0	1120	0,068
1/7	3214	0,196
2/7	4850	0,296
3/7	3425	0,209
4/7	1995	0,122
5/7	784	0,048
6/7	541	0,033
1	455	0,028
Total	16384	1

Figura 9 - Exemplo de Histograma.

Fonte: Marques Filho e Vieira Neto, 1999.

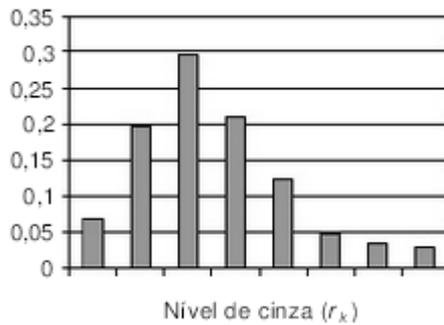


Figura 10 - Exemplo de histograma para imagem com oito níveis de cinza.

Fonte: Processamento de Imagens Digital, 1999.

A Figura 11 apresenta uma imagem com peças de lego para simulação da escala de cinza e histograma.

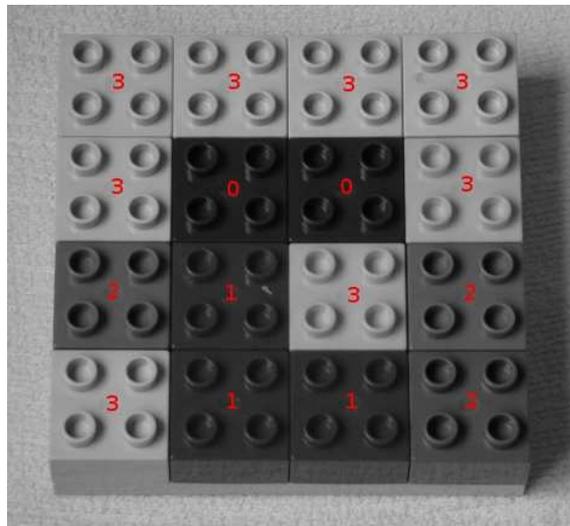


Figura 11 - Exemplo de escala de Intensidade.

Fonte: IMAGE SURVEY, 2013.

Como comentado anteriormente, a cor preta é a ausência de cores. Analisando a Figura 11, percebe-se que o bloco de cor preta recebe o valor zero. Conforme decresce a intensidade da cor, os valores crescem. O histograma é uma técnica que permite a representação de dados, podem ser verificados os valores de frequência, ponto central, variação, amplitude e assimetria. A Figura 12 apresenta o histograma da imagem da Figura 11. Nota-se que o número de peças nas colunas de cada cor da Figura 12 é o mesmo número de cada bloco de cor da Figura 11. Ou seja, o histograma apresenta pelo valor do eixo da Frequência, quantos *pixels* apresentaram o nível digital relacionado no eixo da Intensidade da Cor.

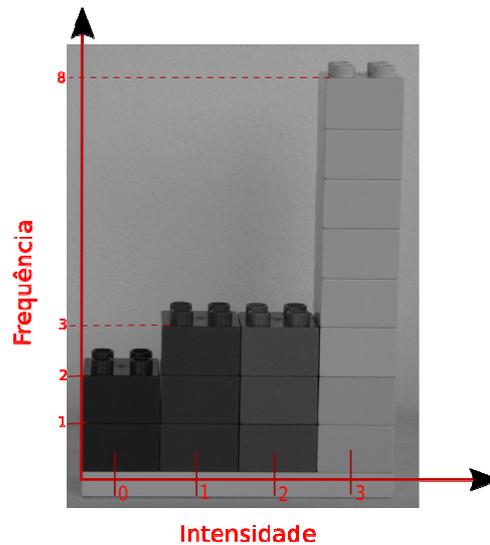


Figura 12 - Intensidade e Frequência da Figura 11.

Fonte: IMAGE SURVEY, 2013.

De acordo com Conci (2008), o histograma, quando aplicado em imagens coloridas, pode ser representado pela quantidade cores que são encontradas na imagem. A Figura 13 apresenta uma imagem colorida dividida em três imagens distintas, ou seja, divididas nos três canais de cores RGB.



(a) Imagem Colorida



(b) Banda Vermelha (Red)



(c) Banda Verde (Green)



(d) Banda Azul (Blue)

Figura 13 - Imagem colorida e imagens com as bandas RGB.

Fonte: CONCI et al, 2008.

A Figura 14 apresenta os histogramas de cada banda monocromática das imagens apresentadas na Figura 13.

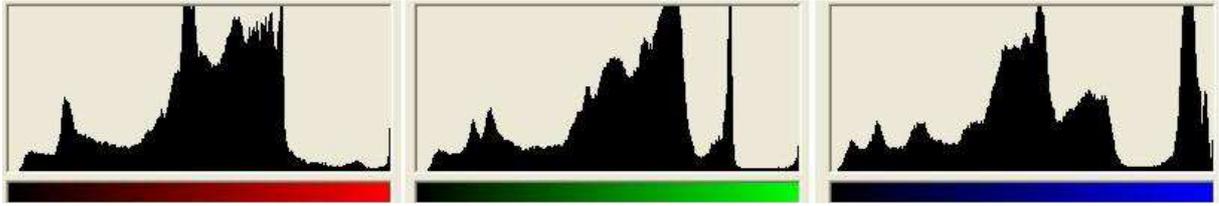


Figura 14 - Histograma da imagem da Figura 13, separadas nas bandas do sistema de cores RGB.

Fonte: CONCI et al, 2008.

Após os conceitos apresentados relacionados a Processamento Digital de Imagens, a seguir serão apresentadas informações sobre as tecnologias que foram aplicadas no desenvolvimento do presente trabalho.

2.2 – TECNOLOGIAS WEB

2.2.1 – OPENCV

A biblioteca OpenCv (*Open Source Computer Vision Library*) foi desenvolvida originalmente pela Intel, tendo seu uso livre para desenvolvimentos de aplicativos da área de Visão Computacional. Essa biblioteca foi criada com a intenção de tornar a visão computacional mais acessível aos usuários, programadores e a interação humano computador em tempo real e a Robótica. A biblioteca OpenCv está dividida em cinco grupos de funções: Processamento de Imagens, Análise estrutural; Análise de movimento e *rastreamento* de objetos; Reconhecimento de padrões e Calibração de câmera e reconstrução 3D.

2.2.2 - SMARTPHONE

O smartphone é um celular com tecnologias avançadas, geralmente com processador e sistema operacional equivalente a sistemas computacionais normais. Eles possibilitam que as pessoas sejam capazes de realizar diversas atividades por meio do aparelho celular, assim não é preciso carregar um computador para realizar atividades rotineiras como ler e-mails, tirar fotografias, pagar contas, pesquisar

informações na internet, etc.

2.2.3 - HTML5

HTML5 é um novo padrão de linguagem de marcação de hipertexto que surgiu no mercado. A sua versão anterior, 4.01, foi colocada em produção no ano de 1999. Mesmo estando em testes e constante desenvolvimento, a maioria dos navegadores já dá suporte às *tags* e API's do HTML5. Como vantagens desta tecnologia, podem ser citados:

- Novos recursos deve ser baseada em HTML, CSS, DOM e JavaScript;
- Reduzir a necessidade de *plugins* externos (como o Flash);
- Melhor tratamento de erro;
- Mais de marcação para substituir o script;
- HTML5 deve ser independente do dispositivo;
- O processo de desenvolvimento deve ser visível para o público;

Algumas informações sobre as *tags* do HTML5 são apresentadas a seguir:

-TAG CANVAS:

O elemento `<canvas>`, é utilizado para criar desenhos, em tempo real, em uma página web. Para isso é utilizado o *scripting*, sendo normalmente utilizada a linguagem *JavaScript*. Além disso, esse elemento é um recipiente para gráficos. As Figuras 15 e 16 apresentam um exemplo dessa aplicação.

```
Código-Fonte  Histórico
<!DOCTYPE html>
2 <html>
3 <body>
4
5 <canvas id="TesteCanvas" width="100" height="50" style="border:4px solid blue;">
6 Seu navegador não suporta o HTML5.</canvas>
7
8 <script>
9
10 var c=document.getElementById("TesteCanvas");
11 var ctx=c.getContext("2d");
12 ctx.font="18px Arial";
13 ctx.fillText("Meu TCC",10,50);
14
15 </script>
16
17 </body>
18 </html>
```

Figura 15 - Código fonte da utilização da tag<canvas>.

Fonte: Autoria própria.

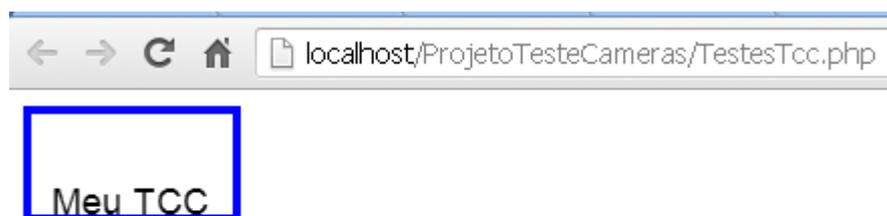


Figura 16 - Resultado da exemplificação tag<canvas>.

Fonte: Autoria própria.

-TAG VIDEO E AUDIO:

Vídeo: Hoje em dia não existe padrão para apresentar vídeos e filmes em uma página web. A maioria dos vídeos é mostrada com a aplicação de algum plugin, como exemplo, o *Flash Player*. No entanto, os navegadores podem ter *plugins* diferentes para a execução de vídeos.

```
html body div
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <link rel="stylesheet" type="text/css" href="galler/css/style.css" />
5 <script type="text/javascript" src="galler/js/vars.js.php"></script>
6 <script type="text/javascript" src="galler/js/json.js"></script>
7
8 </head>
9 <body>
10 <div id="centerDiv">
11 <video autoplay></video>
12 <div><i>Clique para capturar a imagem</i></div>
13 </div>
14 </body>
15 </html>
```

Figura 17 - Código Fonte Utilizando a tag <Video>.

Fonte: Autoria Própria.

```
if(navigator.getUserMedia) {
    navigator.getUserMedia('video', successCallback, errorCallback);
    function successCallback( stream ) {

        video.src = stream;
        video.width = screen.width;
        video.style.border = '1px solid red';
        video.onclick = function(e) {
            }
    }
}
```

Figura 18 - JavaScript para execução da tag<vídeo>.

Fonte: Autoria Própria.

As Figuras 17 e 18 apresentam exemplos de código com uso da tag Vídeo do HTML5 que foram implementados no presente trabalho. A Figura 20 apresenta a interface web desenvolvida que permite a captura de imagens por meio do navegador.

Fonte: Autorial Própria.

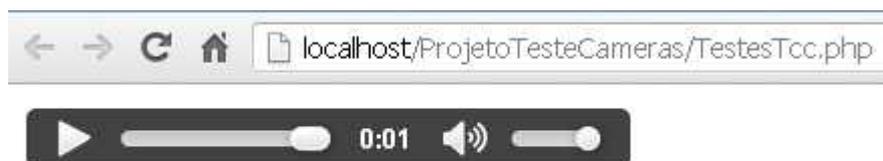


Figura 21 - Reprodutor de audio gerada com a tag<Audio>.

Fonte: Autorial Própria.

- ARMAZENAMENTO LOCAL:

Com o armazenamento local do HTML5, inicia-se a era da *Web Semantica*, porque os objetos criados permitem:

Local Storage: Armazena os dados de navegação sem data para expirar, ou seja, mesmo que o usuário encerre o navegador, as informações contidas nesse objeto não serão apagadas.

Session Storage: Ao contrário do anterior, esse objeto somente armazena os dados pelo período de tempo de uma sessão. Desse modo, quando o usuário fecha o navegador, os dados são apagados.

2.2.5 – JAVASCRIPT

JavaScript é uma linguagem de programação, que é executada na página do cliente, melhorando a interatividade com o usuário. Essa linguagem foi criada pela Netscape em 1995, a princípio visando atender as necessidades com interação do cliente com a página *web*, principalmente na validação de dados de formulários.

Essa linguagem é dinâmica, de modo que as variáveis não são declaradas no início do programa. Além disso, a mesma é interpretada, ou seja, os códigos são interpretados em tempo real e executados pelo navegador no momento em que o usuário utiliza a página *web* em seu navegador. Para que seja criado um *script*, é preciso de um editor de texto como o bloco de notas, e assim iniciar as *tags* `<script></script>`.

A Figura 26 apresenta a implementação do código do presente trabalho utilizando JavaScript.

```
<script>
function checkScreen(event){
    if((screen.width > screen.height) && (mode==0)){
        mode = 1;
        video.videoWidth = screen.height;
        video.videoHeight = screen.width;
    }
    if((screen.width < screen.height) && (mode==1)){
        mode = 0;
        video.videoWidth = screen.width;
        video.videoHeight = screen.height;
    }
}
</script>
```

Figura 22 - Utilização do JavaScript.

Fonte: Autoria Própria.

2.2.4 - AJAX

“A palavra Ajax vem da expressão *Asynchronous JavaScript and XML*. É o uso sistemático de *JavaScript* e *XML*(entre outras tecnologias) para tornar o navegador mais interativo com os usuários, utilizando-se solicitações assíncronas de informações”(NIEDERAUER, 2007).

“AJAX não é uma linguagem nova, mas uma nova maneira de utilizar as linguagens existentes para criar melhores, mais rápidas e interativas aplicações web. AJAX tem por objetivo trocar informações entre servidor e cliente, e manipular o conteúdo da página sem ter que recarregá-la”(REFSNES; HENTHORNE, 2010).

Usando AJAX, as páginas e a aplicação apenas solicitam ao servidor o que realmente precisam apenas as partes que precisam mudar, os dados que o servidor precisa fornecer. Isso significa menos tráfego, menos atualizações e menos tempo esperando uma página ser recarregada (RIORDAN, 2008).

Para utilizar AJAX é preciso ter conhecimento com Javascript, XML e HTML. A partir do objeto XMLHttpRequest do Javascript é possível fazer a troca de informações entre o cliente e o servidor, independente da linguagem de programação utilizada no servidor (REFSNES; HENTHORNE, 2010).

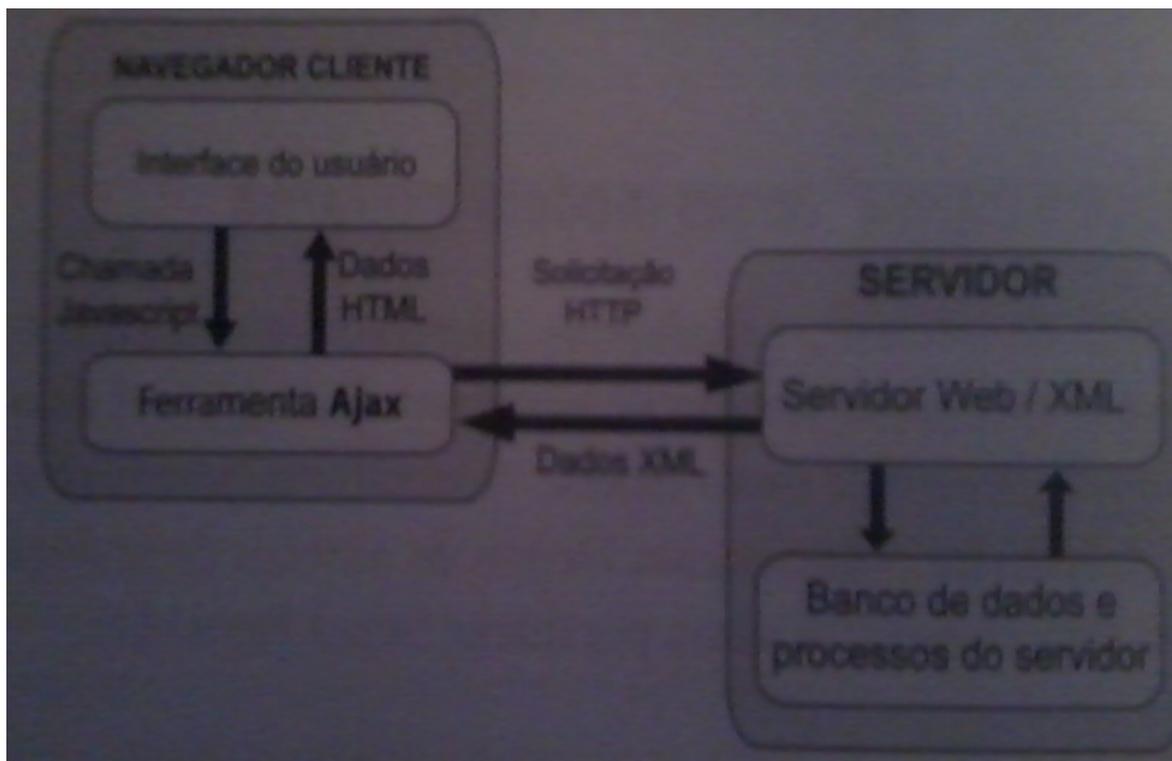


Figura 23 - Modelo de aplicação Ajax.

Fonte: Niederauer, Juliano 2007.

A Figura 24 mostra a interação entre o navegador e o servidor. Percebe-se que a comunicação entre os dois não é direta, intermediando a comunicação, a Ajax realiza seu papel. Como o Ajax é acionado via JavaScript, o usuário poderá visualizar a página normalmente enquanto o mesmo faz a comunicação com o servidor.

Quando o servidor receber algum dado da página cliente, o mesmo irá processar essa mensagem e poderá exibir algum retorno, esse retorno também passa pelo Ajax, assim sendo passado para a página do cliente. Essa passagem não atrapalha na visualização do cliente, pois se ela estiver esperando um retorno, o mesmo vai ser passado somente em uma parte da página, deixando o restante da página inalterado.

Dessa maneira, nota-se que a interação cliente servidor com o Ajax traz agilidade e retira redundâncias de envio de mensagens para o servidor, conseqüentemente melhorando o tráfego da rede e a usabilidade de páginas *web*.

Para que sejam enviados ou recebidos dados do servidor para a página cliente, necessita-se que exista um formulário em HTML. Neste formulário existirá um *link* ou um botão com a atribuição de um *submit*. Quando o usuário realizar um clique com o mouse, as informações serão enviadas para o servidor. Assim, serão processadas e poderá ser repassada ao usuário uma resposta. Enquanto o servidor processa esses dados, o usuário é obrigado a esperar até que a informação esteja pronta para ser retornada e recarregar a página (REFSNES; HENTHORNE, 2010).

Com a utilização da ferramenta AJAX, o javascript comunica-se com o servidor através do objeto XMLHttpRequest. Assim que uma página web cliente consegue se comunicar com o servidor e receber uma resposta sem precisar ser recarregada. Continuando na mesma página o usuário nem perceberá que a informação foi processada por um servidor, dando a impressão de que a aplicação web está interagindo com o usuário da mesma maneira que uma aplicação desktop (REFSNES; HENTHORNE, 2010).

– Aplicação AJAX

A Figura 25 apresenta a implementação do código do presente trabalho utilizando a tecnologia AJAX.

```
function StartAjax(){
    if(window.XMLHttpRequest){
        ajax = new XMLHttpRequest();
    }else if (window.ActiveXObject){
        ajax = new ActiveXObject("Msxml2.XMLHTTP");//Opera Firefox
        if(!ajax){
            ajax = new ActiveXObject("Microsoft.XMLHTTP");// Internet Explorer
        }
    }
    else{
        alert("Use OPERA como navegador, esse sim! Funciona!");
    }
    return ajax;
}
```

Figura 24 - Implementação usando Ajax.

Fonte: Autoria Própria.

No trecho de código da Figura 25 mostra-se a chamada da função *StartAjax*, a qual verificar qual navegador está sendo utilizado pelo usuário. Esse método resolve problemas de incompatibilidades entre sistemas web e navegadores.

2.2.6 – JQUERY

JQuery ficou popular entre os desenvolvedores de aplicação web por causa de sua simplicidade e facilidade de uso. A biblioteca possui suporte de uma comunidade de desenvolvedores e tem crescido significativamente ao longo dos anos, desde a sua criação, em 2006, por John Resig (ALLANA, 2011).

Com JQuery fica fácil manipular os elementos das páginas, adicionar e modificar atributos HTML e propriedades CSS, definir eventos e animações. Essa biblioteca também possui suporte a AJAX para fazer dinamicamente HTTP requests e outras funcionalidades gerais. Por estar sendo bastante utilizado, os desenvolvedores devem estar familiarizados com o JQuery, mesmo que não a utilizem no seu código, é bem provável que irá encontrar no código de outras pessoas (FLANAGAN, 2011).

Para utilizar o JQuery na página web cliente, é preciso referenciar a biblioteca com a chamada do script, apresentada na Figura 27.

```
<script type='text/Javascript ' src='jquery-1.4-min.js'></script>
```

Figura 25 - Chamada da biblioteca jquery.

Fonte: Autoria própria.

Um exemplo de utilização de JQuery é mostrado na Figura 28. Nesta Figura é demonstrado o momento do evento de clique em uma *tag*<p>,

```
// Ao clicar em qualquer elemento <p> muda-se a cor de fundo do elemento  
$("p").click(function() {  
$(this).css("background-color", "gray");  
});
```

Figura 26 - Utilização de jquery.

Fonte: Desenvolvimento em ASP.NET MVC, JQUERY E AJAX, Cherubini, Negir Eivaldo 2011.

2.2.7 – FLASH LITE

A Adobe *Flash* é uma plataforma pertencente à empresa Adobe System Incorporate. É uma boa alternativa para o desenvolvimento de projetos da área web para dispositivos móveis. A empresa é uma das pioneiras no desenvolvimento web com características de aplicações *desktop*, sendo que o Flash inovou o desenvolvimento, com aplicações que rodam direto no navegador do usuário.

Flash Lite é uma tecnologia projetada para dispositivos móveis, que busca aliar a performance do aparelho aos recursos presentes no mesmo. Ainda, o Flash Lite facilita o trabalho de profissionais que já desenvolviam conteúdos em Flash para ambiente *desktop*. Isto porque, tais profissionais podem utilizar o conhecimento adquirido para realizarem trabalhos para dispositivos móveis. (LR de Oliveira ¹, RD Medina ¹ - 2007).

O Flash Lite permite que desenvolvedores com pouca experiência criem interfaces em pouco tempo, desta maneira não necessitando de uma carga grande de conhecimento para o desenvolvimento.

2.2.8 – PHP

A primeira versão do PHP surgiu em 1995, criada por Rasmus Lerdorf para uso pessoal. Com o passar dos anos, essa linguagem se tornou uma das mais utilizadas na atualidade. Sendo que sua portabilidade é uma das maiores vantagens do PHP, possibilitando sua instalação em vários Sistemas Operacionais como: Windows, Linux, Unix, IBM iSeries, SGI IRIX, RISC OS, Netware Novell, Mac OS X e AmigaOS

O PHP é uma linguagem com um modelo de desenvolvimento muito simples. O objetivo inicial do PHP era tornar possível um desenvolvimento rápido sem um treinamento preliminar. Esse objetivo foi alcançado, fazendo com que a maioria das empresas de hospedagem ofereçam PHP nos seus servidores.

Com essas e demais facilidades encontradas no PHP, determinamos está linguagem para desenvolver a parte servidor.

3 – METODOLOGIA

Nesse capítulo será apresentada a metodologia adotada para o desenvolvimento do aplicativo para processamento de imagens capturadas por *smartphones*.

3.1 – Cliente – Smartphone

As seguintes limitações foram analisadas:

- Limitações de Processamento: É possível perceber que os celulares tem um processamento abaixo do que um computador *desktop*. Consequentemente, essa limitação impede o desenvolvimento de aplicações que exijam processamento rápido e com grande volume de transmissões de dados;

-Limitações de Memórias: a baixa capacidade de memória RAM dos celulares torna difícil a alocação de espaço para o uso de aplicativos pessoais.

-Limitações de Software: Com relação ao software, existem restrições aos sistemas operacionais de cada aparelho *smartphone*, podendo até dificultar o uso de recursos específicos.

Analisando as limitações comentadas anteriormente, pode-se dizer que os usuários de dispositivos móveis são poucos tolerantes a espera para resposta de informações, sendo que *smartphones* não apresentam tempo de inicialização de um sistema comum. Ou seja, os usuários são impacientes com a demora dos softwares assim precisamos que o software tenha um desempenho considerável na sua execução e na sua resposta. Para que seja possível o desenvolvimento de uma aplicação adequada, é preciso escolher bem a plataforma, sempre considerando os fatores de mobilidade e as expectativas dos usuários.

A plataforma Flash Lite, pertencente a Adobe System Incorporate, utilizada para o desenvolvimento de sistemas para os *smartphones*, apresenta alguns pontos negativos a serem considerados:

- Indisponibilidade: há poucos *smartphones* com compatibilidade ao Flash Lite, principalmente no Brasil;

-Inconsistência de versões: em determinados aparelhos e sistemas operacionais, a plataforma Flash Lite não é suportada integralmente;

-Política de Distribuição: quando os smartphones saem de fábrica sem o suporte a Flash Lite, os usuários devem pagar cerca 10 (dez) dólares para que possam ter o suporte no seu aparelho.

Partindo disso, para o desenvolvimento do projeto foi utilizado o HTML5. Foi utilizada a tag de vídeo para ser possível capturar a imagem da câmera do celular nos navegadores. Foi utilizado o navegador Opera, porque o mesmo foi o que teve melhor desempenho nos testes realizados neste projeto. Mesmo que o HTML5 ainda esteja em fase de desenvolvimento, segundo o site do navegador Opera , setenta e cinco(75) per cento das compatibilidades entre Opera e HTML5 já estão feitas.

A Figura 29 apresenta a pagina HTML do cliente desenvolvido.

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4     <link rel="stylesheet" type="text/css" href="gallery/css/style.css" />
5     <script type="text/javascript" src="gallery/js/vars.js.php"></script>
6     <script type="text/javascript" src="gallery/js/json.js"></script>
7     <script type="text/javascript" src="gallery/js/jquery-1.7.2.min.js"></script>
8 </head>
9 <body>
10     <div id="centerDiv">
11         <video autoplay></video>
12         <div><i>Clique para capturar a imagem</i></div>
13         <script type="text/javascript" src="gallery/js/jsscripts.js"></script>
14     </div>
15 </body>
16 </html>
```

Figura 27 - Código Cliente HTML5 para chamada da Câmera do smartphone.

Fonte: Autoria Própria.

Percebe-se que na linha um(1) está sendo feita a declaração inicial do HTML5. Nas linhas que estão entre a tag <head>, estão os arquivos que serão utilizados para a formatação da página, gravação e manipulação de dados.

Dentro da tag<body>, existe uma outra tag chamada <video> que é suportada pelo seguintes navegadores: Internet Explorer 9 +, Firefox, Opera, Chrome e Safari . E essa tag faz a chamada para a câmera que esta acoplada ao smartphone.

Para ser possível a captura da imagem, utiliza-se a linguagem JavaScript como mostrado na Figura 30.

```

function successCallback( stream ) {

    video.src = stream;
    video.width = screen.width;
    video.style.border = '1px solid red';
    video.onclick = function(e){

    }

}

function checkScreen(event){
if((screen.width > screen.height) && (mode==0)){
    mode = 1;
    video.videoWidth = screen.height;
    video.videoHeight = screen.width;
}
if((screen.width < screen.height) && (mode==1)){
    mode = 0;
    video.videoWidth = screen.width;
    video.videoHeight = screen.height;
}
}
}

```

Figura 28 - JavaScript trabalhando com o elemento vídeo do HTML5.

Fonte: Autoria Própria.

Feito isso, captura-se a imagem, a qual está sendo disponibilizada no navegador do cliente. Além disso, pode-se deixar a câmera com um tamanho adequado para a resolução do aparelho. Quando realiza-se um clique na imagem, a captura é realizada pelo método vídeo.onclick, dessa maneira também é feito o armazenamento da imagem capturada no banco de dados.

3.2 – Processamento de Imagem – PHP

Para o processamento de imagens é utilizada a linguagem PHP, desse modo sendo possível extrair a cor predominante da imagem pela parte cliente da aplicação.

Por meio do Jquery é feita uma requisição assíncrona (Ajax) para o servidor, passando como parâmetro o signo escolhido pelo usuário e a imagem (convertida em base 64). Dentro do código Ajax, o primeiro passo realizado é a inserção a imagem passada por parâmetro em base 64. Após a conversão da imagem para

uma variável do tipo *String*, é feito um recorte. Então, converte-se novamente a imagem para uma paleta de 64 cores.

No processo de análise da imagem é utilizado um laço de repetição, com o qual se percorre toda a matriz de pixels da imagem. Em cada pixel é extraído o valor das cores no sistema RGB.

Também é realizado uma requisição HTTP, para um determinado site de Astrologia popular no Brasil (www.joaobidu.com.br), o qual contém as previsões das cores para todos os signos diariamente.

Após a consulta ao site, os valores são agrupados de acordo com a representação hexadecimal da cor do sistema RGB. Nesse agrupamento são somados todos os valores hexadecimais em comum, para que seja contabilizado a cor com maior ocorrência, desse modo caracterizando a predominância da mesma na imagem.

Na sequência do processo, utiliza-se uma tabela de equivalência de cores, a qual possui tanto os valores hexadecimais quanto os nomes das cores. Neste ponto foram adotadas as seguintes cores: branco, preto, cinza, vermelho, azul claro, azul, azul escuro, verde claro, verde, verde escuro, amarelo claro, amarelo, amarelo escuro, marrom claro, marrom, marrom escuro, roxo claro, roxo, roxo escuro, laranja claro, laranja, laranja escuro, rosa claro, rosa e rosa escuro.

Essa equivalência entre hexadecimal e nome de cor foi realizada para que seja possível comparar o valor extraído da imagem com a cor do signo adquirida para o dia corrente.

4- RESULTADOS

4.1 - EXPERIMENTO 1

A Figura 31 apresenta a interface desenvolvida para a captura da imagem do usuário, para obter os resultados abaixo foi utilizado um notebook com sua webcam para a captura das imagens.. Além da captura, o resultado da análise da imagem pode ser visto na mesma interface.



Figura 29 - Experimento 1.

Fonte: Autoria Própria.

4.2 – EXPERIMENTO 2

Infelizmente, devido a resolução do aparelho smartphone e a iluminação ambiente, a cor extraída foi preta. Visualmente, percebe-se que a cor da blusa era roxa.



Foto Tirada



Area Analisada



Cor Predominante

ROXO

Cor no Horócopo

CINZA

Figura 30 - Experimento 2.

Fonte: Autoria Própria.

Analisando a Figura 32, percebe-se visualmente que a blusa utilizada é roxa. De acordo com o signo analisado, o horóscopo indicava a cor cinza. Desse modo, o sistema acertou a cor predominante e indicou a diferença entre as cores.

4.3 – EXPERIMENTO 3



Figura 31 - Experimento 3.

Fonte: Autoria Própria.

Novamente, devido a resolução e iluminação, a imagem capturada não foi totalmente fiel a imagem real. Com a pouca incidência de luz, o sistema definiu que a cor predominante era o roxo, porém a verdadeira cor era cinza.

4.4 – EXPERIMENTO 4



Figura 32 - Experimento 4.

Fonte: Autoria Própria.

No teste número 4 (Figura 34), o sistema teve acerto. A camiseta do aluno fotografado era visualmente preta. Devido a boa iluminação, a imagem foi consistente com a realidade.

4.5 – EXPERIMENTO 5



Figura 33 - Experimento 5.

Fonte: Autoria Própria.

No teste apresentado na Figura 35, novamente ocorre erro, devido a qualidade da imagem capturada

4.6 – EXPERIMENTO 6



Foto Tirada



Area Analisada



Cor Predominante

CINZA

Cor no Horócopo

AZUL-CLARO

Figura 34 - Experimento 6.

Fonte: Autoria Própria.

Na Figura 36 é percebido que existe alta incidência de luz na imagem. Desta maneira, o sistema pode analisar corretamente a imagem, determinando que sua cor predominante é o Cinza.

4.7 – EXPERIMENTO

Color Camera

Áries

Clique para capturar a imagem



Foto Tirada

Area Analisada

Cor Predominante
AZUL CLARO

Cor no Horócopo
AZUL-CLARO

Figura 35 - Experimento 7.

Fonte: Autoria Própria.

Na Figura 35 é possível verificar que a luz fez que uma camiseta de cor branca tornasse de cor azul claro, sendo que a última cor foi processada pelo aplicativo.

5 - CONCLUSÕES E PERSPECTIVAS DE PESQUISAS FUTURAS

O objetivo deste trabalho foi extrair a cor predominante de roupas em imagens de smartphones. Inicialmente, pretendia-se usar a biblioteca OpenCv. No entanto, após inúmeras tentativas de processamento, verificou-se que a OpenCv não estava atendendo os requisitos que foram impostos no sistema. Assim, foi utilizado apenas a linguagem PHP para a extração de cores predominantes de imagens, atendendo as expectativas propostas no sistema.

Na fundamentação teórica foi comentado que quando se trabalha com cores , trabalha-se com luz. Pode ser observado nos resultados apresentados que houve imagens capturadas com pouca incidência de luz. Essas imagens foram processadas erroneamente por causa da qualidade da imagem. Todavia, quando havia iluminação adequada, o aplicativo conseguiu extrair a cor exata da imagem.

Como indicação de trabalho futuro, indica-se a utilização de filtros para correção da iluminação das imagens capturadas.

6 - REFERENCIAS

ALLANA, Sonal Aneel. **ASP.NET JQuery Cookbook**. Packt Publishing. Birmingham – Mumbai, 2011.

CASTAGNETTO, Jesus et Al. **Professional PHP: programando**. São Paulo: Makron, 2001.

CASTLEDINE, Earle; SHARKIE, Craig. **JQuery: Novice to Ninja**. SitePoint Pty, Estados Unidos. Fevereiro, 2010.

CHERUBINI, Negir Edivaldo. **Desenvolvimento em ASP.NET MVC, JQYERY e AJAX**, 2011. 55. Trabalho de Conclusão de Curso(Tecnólogo em Análise e Desenvolvimento de Sistemas). UTFPR, Medianeira, Paraná.

CONCI, Aura; AZEVEDO, Eduardo; LETÃ, R. Fabiana (2008). **Computação Gráfica**. Rio de Janeiro: Elsevier, 2008.

Desvendando o HTML5 - Fatos e mito. Disponível em<<http://my.opera.com/operabrasil/blog/2010/06/11/desvendando-o-html5-fatos-e-mitos-2>> Acessado em: 10/03/2013.

FLANAGAN, David. **JQuery Pocket Reference**. O'Reilly Media, 2011.

GONZALEZ, C. Rafael; Woods, E. Richard. **Processamento de Imagens Digitais**. São Paulo: São Paulo, 2000.

IMAGE SURVEY. Disponível em <http://www.imagesurvey.com.br/2010/11/histogramas-de-imagem/> Acesso em 15/01/2013.

Histogramas – Luminância e Cor em Histogramas. Disponível em:< <http://www.cambridgeincolour.com/pt-br/tutorials/histograms2.htm>>. Acessado em:05/03/13.

JUNIOR, Silva da Maia Jucima. Firmino, M. Carlos Emiliano. **Desenvolvimento de Jogos em HTML5**. SBC - Proceedings of SBGames 2010. P. 268-271.

NIEDERAUER, Juliano. **Web interativa com Ajax e PHP**. São Paulo: Novatec, 2007.

KEITH, Jeremy. **Bulletproof Ajax**. New Riders – USA, 2007.

MARQUES FILHO, Ogê; VIEIRA NETO, Hugo. *Processamento Digital de Imagens*, Rio de Janeiro: Brasport, 1999

MUTO, Cláudio Adonai. **PHP & MySQL: guia introdutório**. 2. ed. Rio de Janeiro: Brasport, 2004.

O que é HTML5 . Disponível em:< <http://www.tecmundo.com.br/navegador/2254-o-que-e-html-5-.htm>> Acesso em:21/02/2013.

OLIVEIRA, Ramos de Leandro. MEDINA, Duarte Roseclea. **Desenvolvimento de objetos de aprendizagem para dispositivos móveis: uma nova abordagem que contribui para a educação.** Centro de Tecnologia – Universidade Federal de Santa Maria (UFSM).

PARQUE, Yangil. CHEN, V. Jenchung(2007). **Aceitação e adoção do uso inovador de smartphone.** Trabalho de Pesquisa.Industrial Management & Data Systems Vollss 107:9, PP.1349-1365.

Pensamentos Sobre Flash. Disponível em <<http://www.apple.com/hotnews/thoughts-on-flash/>> Acesso em : 01/03/2013.

PILGRIM, Mark. **HTML5 Up and Running.** P. 81 -85, August 2010.

REFSNES, Hege, Stale, Kai Jim e Jan Egil; HENTHORNE, Kelly. **Learn Javascript and Ajax with w3schools.** Wiley Publishing, Inc. Indianapolis, Indiana – 2010.

RIORDAN, Rebecca M. **Head First Ajax.** O'Reilly Media, Inc. Estados Unidos, 2008.

ROCHA, Carlos João. **COR LUZ, COR PIGMENTO E OS SISTEMAS RGB E CMY,** 2010.

SANTOS, Moraes dos Renato. **UM ESTUDO DE PROCESSAMENTO DE IMAGENS COM OPENCV,** 2011. Trabalho de Conclusão de Curso(Tecnólogo em Sistema da Informação). Universidade Federal Fluminense, Niterói, Rio de janeiro.

SCURI, Escanõ Antonio. **Fundamentos da Imagem Digital.** Tecgraf/PUC, Janeiro de 2009.

TABORBA, Boavista Nuno João. SILVA, Dias André Tiago. **Reconhecimento de Objetos Simples**(2005). Trabalho realizado no âmbito de inteligência artificial do 2º semestre, do 3º ano, da Licenciatura em Engenharia Informática e Computação da Faculdade de Engenharia da Universidade do Porto. Porto, Portugal.

VALERIANO, A. Allan. ENDLER, Markus. TrackService: **Criação, compartilhamento e avaliação de Track Logs em tempo real usando smartphones.**SBCUP-2009.