



UNIVERSIDADE FEDERAL DO MARANHÃO
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA
CURSO DE GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

JULLYANA FIALHO PINHEIRO

**UMA METODOLOGIA PARA DETECÇÃO DE ESQUEMAS TÁTICOS EM
PARTIDAS DE FUTEBOL**

SÃO LUÍS

2016

JULLYANA FIALHO PINHEIRO

UMA METODOLOGIA PARA DETECÇÃO DE ESQUEMAS TÁTICOS EM PARTIDAS DE
FUTEBOL

Monografia apresentada ao Curso de Ciência da
Computação da Universidade Federal do Ma-
ranhão, como parte dos requisitos necessários
para obtenção do grau de bacharel em Ciência
da Computação.

Orientador: Prof. Dr. Geraldo Braz Jr.

SÃO LUÍS

2016

Pinheiro, Jullyana Fialho

Uma Metodologia para Detecção de Esquemas Táticos em Partidas de Futebol/ Jullyana Fialho Pinheiro. – São Luís, 2016.

47f.

Orientador: Geraldo Braz Jr

Monografia (Graduação) – Universidade Federal do Maranhão, Curso de Ciência da Computação, 2016.

1. Futebol 2. Táticas 3. Processamento de Imagens 3. Aprendizado de Máquina

CDU 004.92:796.332

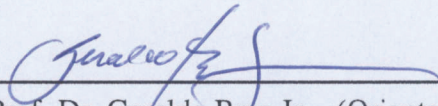
JULLYANA FIALHO PINHEIRO

UMA METODOLOGIA PARA DETECÇÃO DE ESQUEMAS TÁTICOS EM PARTIDAS DE
FUTEBOL

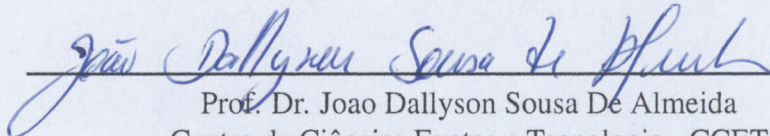
Monografia apresentada ao Curso de Ciência da
Computação da Universidade Federal do Ma-
ranhão, como parte dos requisitos necessários
para obtenção do grau de bacharel em Ciência
da Computação.

Aprovada em: 04 de Abril de 2016

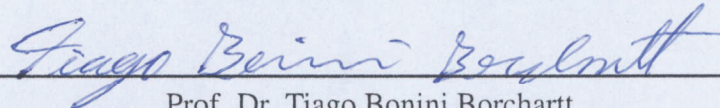
BANCA EXAMINADORA



Prof. Dr. Geraldo Bráz Jr. (Orientador)
Centro de Ciências Exatas e Tecnologia - CCET
Universidade Federal do Maranhão - UFMA



Prof. Dr. Joao Dallyson Sousa De Almeida
Centro de Ciências Exatas e Tecnologia - CCET
Universidade Federal do Maranhão - UFMA



Prof. Dr. Tiago Bonini Borchart
Centro de Ciências Exatas e Tecnologia - CCET
Universidade Federal do Maranhão - UFMA

Agradecimentos

À minha mãe, minha heroína, por ter estado sempre do meu lado em todos os momentos e ser minha grande inspiração.

Ao meus irmãos, Gustavo e Isabella, por serem meus companheiros de vida.

Ao meu namorado Jordan, pelo apoio incondicional, por me ensinar a ser uma pessoa melhor e por viver essa aventura comigo.

Aos meus tios, pelos ensinamentos e amor que sempre recebi.

À minha madrinha, Enide, por ter me ensinado que a leitura é uma das maiores fontes de felicidade e conhecimento.

Ao meu orientador, Geraldo Braz Junior, pela paciência e pelos conhecimentos passados, não só na realização deste trabalho, mas durante toda a graduação.

Resumo

O futebol é um dos esportes mais praticados no mundo e suas competições possuem espectadores em todos os continentes, tornando popular pesquisas que visam entender o que ocorre em campo durante seus jogos. Um dos fundamentos mais importante deste esporte é a tática. Tática compreende desde a função exercida por cada jogador, a sua movimentação e o seu posicionamento. A metodologia apresentada neste trabalho propõe o reconhecimento da formação tática, ou seja, o posicionamento adotado por cada jogador durante uma partida de futebol. Utilizando de técnicas de processamento de imagem, como filtros lineares, operações morfológicas e o detector de bordas de Canny, para segmentação dos *frames*; aprendizado de máquina com o algoritmo *K-Nearest-Neighbor* para classificação dos times; e conhecimentos específicos do esporte para a definição da formação. A metodologia apresentada obteve acurácia de 61.6% na detecção da formação.

Palavras-chaves: Futebol. Táticas. Processamento de Imagens. Aprendizado de Máquina.

Abstract

Soccer is one of the most played sport worldwide and its matches has audience in all continents, which makes popular research aimed to understand what happens during the game. A fundamental piece of this sport is the tatics. Tatics is defined by great number of features such as the function performed by each player, their movement and positioning. The methodology proposed in this paper proposes the recognition of tactical formation, meaning the position taken by each player during the game. The methodology created in this work proposes the tatics recognition of the teams during the games. Using image processing techiniques, as linear filters, morphological operations and Canny edge detector to segment the frames. Machine learning with KNN to classify the teams and specific knowledge on the sport to define the formation. The presented methodology resulted in an accuracy of 61.6 % in the tests.

Keywords: Soccer. Tactics. Image Processing. Machine Learning.

Lista de ilustrações

Figura 1 – Audiência média da copa do mundo de 2014.	12
Figura 2 – Esquemas utilizados na história do futebol.	17
Figura 3 – Organização dos canais em HSV	18
Figura 4 – Exemplo de máscara do filtro da Gaussiana	19
Figura 5 – Representações do <i>kernel</i> Gaussiano	19
Figura 6 – Exemplo - Uso do filtro da Gaussiana	20
Figura 7 – Exemplo - Uso do <i>Median Blur</i>	20
Figura 8 – Exemplo - Uso do Componentes Conectados	22
Figura 9 – Exemplo - Detecção de bordas com Canny	23
Figura 10 – Exemplo - KNN	25
Figura 11 – Visão geral da metodologia desenvolvida.	26
Figura 12 – Exemplo - Extração do Campo em Jogos Diferentes	27
Figura 13 – Uso do <i>Connected Components</i> para obtenção da máscara de campo	28
Figura 14 – Exemplo de uma máscara de campo	28
Figura 15 – Exemplo - Máscara de detecção de jogadores em jogos diferentes	29
Figura 16 – Exemplo - Utilização do Canny no encontro de contornos	30
Figura 17 – Exemplo - Campo com áreas verdes adjacentes	31
Figura 18 – Exemplo - Linhas que forma a divisão do campo em 4 partes	31
Figura 19 – Exemplo - Linhas laterais encontradas com a Transformada de Hough	32
Figura 20 – Exemplo - Linhas projetadas com o uso da divisão horizontal da tela em 4 partes	32
Figura 21 – Exemplo - Separação dos times com base na linha central	33
Figura 22 – Exemplo - Divisão do jogador em duas partes	34
Figura 23 – Exemplo - Funcionamento da matriz de classificação	35
Figura 24 – Exemplo - Linhas de Campo	36
Figura 25 – Exemplo - Uso das Linhas de Campo para Retirada dos Goleiros	37
Figura 26 – Exemplo - <i>Frame</i> descartado devido ao pequeno número de linhas	37
Figura 27 – Fluxograma de seleção das linhas separadoras	38
Figura 28 – Exemplo - Campo onde a máscara gerada apresentou erro	39
Figura 29 – Exemplo - <i>Frames</i> válidos e não válidos	40
Figura 30 – Exemplo - Erro de classificação causado por classificação de times incorreta	41
Figura 31 – Exemplo - <i>Frame</i> classificado de forma correta	41

Lista de tabelas

Tabela 1 – Parâmetros de jogo modificados para obtenção de uma melhor visão do gramado	27
Tabela 2 – Limites de Saturação	27
Tabela 3 – Matriz de Classificação formada a partir da Figura 21	34
Tabela 4 – Formações a serem encontradas	35
Tabela 5 – Porcentagens de acertos referentes a cada formação	41
Tabela 6 – Cálculo de pesos para a formação 3-3-3	42

Lista de abreviaturas e siglas

FIFA	Fédération Internationale de Football Association - Federação Internacional de Futebol
HTTP	Hypertext Transfer Protocol - Protocolo de Transferência de Hipertexto
KNN	K-Nearest-Neighbor - K-Vizinhos Mais Proximos

Sumário

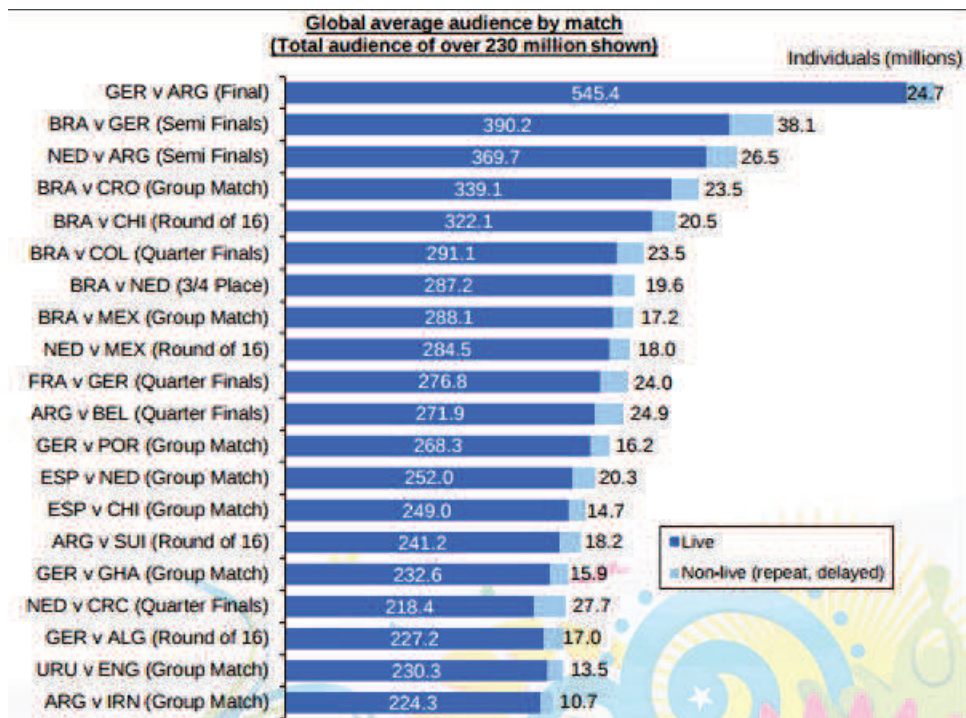
1	INTRODUÇÃO	12
1.1	Objetivo	13
1.1.1	Objetivos Específicos	13
1.2	Trabalhos Relacionados	14
1.3	Organização do Trabalho	14
2	FUNDAMENTAÇÃO TEÓRICA	16
2.1	Esquemas Táticos de Futebol	16
2.2	Técnicas de Processamento de Imagem	16
2.2.1	HSV	16
2.2.2	Thresholding	17
2.2.3	Filtros Lineares	18
2.2.3.1	Suavização Gaussiana	18
2.2.4	Median Blur	19
2.2.5	Operação Morfológica de Fechamento	20
2.2.5.1	Dilatação	21
2.2.5.2	Erosão	21
2.2.5.3	Operação de Fechamento	21
2.2.6	Componentes Conectados	21
2.2.7	Detecção de Bordas Canny	22
2.2.8	A transformada de Hough	23
2.3	Técnicas de Aprendizagem de Máquina	24
2.3.1	K-Nearest-Neighbor	24
3	METODOLOGIA	26
3.1	Captura de Amostras	26
3.2	Extração do Campo	27
3.3	Criação de Máscara de Detecção de Jogadores	29
3.4	Localização dos Jogadores e Remoção de Falso-positivos	29
3.5	Detecção de Times	32
3.5.1	Criação dos Modelos de Times	33
3.5.2	A classificação do KNN	33
3.5.3	Matriz de Classificação	33
3.6	Detecção da Formação	35
3.6.1	Encontro das Linhas Separadoras	36
3.6.2	Busca pelas Formações	38

4	RESULTADOS	39
5	CONCLUSÃO	43
5.1	Trabalhos Futuros	43
	REFERÊNCIAS	44

1 Introdução

O futebol é jogado por 250 milhões de pessoas no mundo, sendo que o Brasil possui o 3º maior número de jogadores registrados, segundo a Federação Internacional de Futebol (FIFA) (FIFA, 2007). A audiência do esporte também é expressiva, no ano de 2014, a Copa do Mundo realizada no Brasil atingiu um total de 2.7 bilhões expectadores (FIFA, 2014). O número médio de pessoas que assistiram ao evento é mostrado na Figura 1 (FIFA, 2014).

Figura 1 – Audiência média da copa do mundo de 2014.



Fonte: FIFA (2014)

Criado na Inglaterra em 1863, o futebol era composto por um conjunto de jogadores, 1 goleiro e 10 jogadores de linha, estes desempenhavam a mesma função, atacar. Com o tempo se percebeu que esses atletas sem atribuições acabavam atrapalhando seus próprios companheiros de equipe. Com isso, funções começaram a ser atribuídas aos jogadores, criando assim os esquemas táticos (PARREIRA, 2005).

Hoje, o futebol tem os esquemas táticos como um dos seus principais alicerces. Tática pode ser entendida como um grupo de regras e comportamentos utilizados para realizar a utilização de forma ótima dos meios condicionais, motores e psíquicos (ROCHA, 2010). No futebol a tática é feita tanto para jogadores de forma individual, como coletiva. Boa parte dela é aplicada com ou sem a posse de bola por parte de um time (GRECO et

al., 2009).

Entender a técnica utilizada por adversários é considerada de extrema importância para se adquirir respostas positivas dentro de um jogo, conseguir observar o esquema usado pelo oponente em partidas anteriores e até mesmo em um tempo anterior da partida atual pode ajudar na definição da tática de um time ou na alteração da mesma (CARVALHO; SCAGLIA; COSTA, 2013).

Um dos sistemas utilizados atualmente para se ter um melhor entendimento do comportamento de jogadores em campo é GPS PRO SOCCER¹, uma ferramenta que utiliza da tecnologia GPS para permitir a análise dos movimentos dos jogadores. Porém o estudo realizado por essa tecnologia tem como objetivo maior obter dados referentes a capacidade física de cada atleta, de forma a melhorar os treinamentos físicos individuais (CAMARÃO, 2009). O problema dessa tecnologia é a falta de viabilidade financeira que ela apresenta para times de pequeno porte. O uso de imagem pode então prover um meio de estudo mais viável do que o utilizado atualmente.

1.1 Objetivo

O objetivo desse trabalho é detectar a formação tática usada por equipes de futebol durante uma partida, utilizando de técnicas de processamento de imagem e aprendizado de máquina, com o intuito de fornecer um meio de análise tática para profissionais e aficionados da área.

1.1.1 Objetivos Específicos

Abaixo estão os objetivos específicos deste trabalho:

- Realizar estudo sobre formação tática no futebol;
- Adquirir base de vídeo/imagem de jogos de futebol;
- Estudar e implementar técnicas de segmentação de imagens para segmentar jogadores
- Estudar e implementar técnicas de reconhecimento de objetos em imagens;
- Estudar e implementar mecanismos de aprendizado de máquina para realizar a classificação de objetos.
- Validar os resultados comparando com o esquema tático oficial do time;

¹ <http://gpsprosoccer.com.br/>

1.2 Trabalhos Relacionados

A maioria dos esportes requer movimentos precisos e rápidos, o que torna difícil sua análise em tempo real, tanto para seus profissionais como para sua audiência. Como o uso de dispositivos eletrônicos nem sempre é possível, não só pela natureza do esporte, mas também pela falta de recursos de agremiações com pouco poder aquisitivo, o uso de visão computacional se torna uma importante ferramenta para esportes em geral (MOESLUND et al., 2014).

No futebol vários trabalhos na área já foram realizados, entre eles, Choi et al. (1997), propuseram uma metodologia para encontrar, identificar e rastrear jogadores em sequencias de imagem, utilizando das informações de cores da imagem, *Kalman Filtering* e modelo de campo.

Khatoonabadi e Rahmati (2009), apresentaram um método para detecção automática de jogadores em cenas de gol, por eliminação do movimento da câmera, que utiliza *Connected Component Labelling* para encontrar os jogadores e *Hough Transform* para detecção das linhas presentes no campo.

A pesquisa desenvolvida por Huang, Llach e Bhagavathy (2007) tem como objetivo detectar e localizar jogadores com o uso de análise baseada na forma, para identificação dos atletas, e encontrar a bola com a extração do plano de fundo, que é obtido com o uso de um histograma de cor do campo e análise de componentes conectados.

Maćkowiak et al. (2010) propuseram um sistema de segmentação de vídeo de transmissões televisivas de jogos de futebol. Os autores desenvolveram um sistema híbrido de segmentação que utiliza uma cor dominante para a extração do campo, detecção de linhas baseada na Transformada de Hough e uma combinação de histogramas orientados a gradiente (HOG) com o classificador Support Vector Machine (SVM) para detectar jogadores.

Neste trabalho a cor dominante também foi utilizada para a realização da extração do campo, assim como a detecção de linhas com a Transformada de Hough, porém enquanto outros trabalhos utilizam comparação de histograma para identificação dos times, o classificador usado nesta metodologia foi o K-Nearest-Neighbor (KNN), com o auxílio de uma matriz de classificação, criada neste trabalho para melhorar os resultados do KNN.

1.3 Organização do Trabalho

Esta monografia está dividida em cinco capítulos. O primeiro contém a introdução do problema, objetivos gerais e específicos.

No segundo capítulo são abordados os conceitos de visão computacional utilizados

neste trabalho, entre eles estão: A transformada de Hough, filtros lineares, operação morfológica de fechamento, entre outros.

No terceiro capítulo estão contidas as metodologias utilizadas ordenadas de acordo com sua implementação.

O quarto capítulo apresenta os resultados adquiridos. Por último, apresentam-se as conclusões obtidas com a realização deste trabalho.

2 Fundamentação Teórica

Neste capítulo são explicados os conceitos e técnicas necessárias para o entendimento da metodologia proposta de reconhecimento de formação tática. Essas técnicas foram divididas por sessões de forma a facilitar a compreensão. Na primeira seção são mostrados as definições táticas referentes a formação da equipe. A fase seguinte é formada por conceitos de processamento de imagens, utilizadas na etapa de segmentação. Na última etapa estão apresentados os métodos de aprendizado de máquina, usados na identificação dos times.

2.1 Esquemas Táticos de Futebol

Os esquemas táticos são formações que definem a posição dos atletas em campo durante uma partida de forma a se obter o melhor resultado possível. Eles são distribuídos de forma a organizar os jogadores em 3 regiões: defesa, meio-campo, ataque. Para dar nome a um sistema, são colocados os números de atletas que compõem cada uma dessas áreas. Um time com 3 defensores, 5 jogadores de meio-campo e 2 de ataque, teria seu sistema definido como 3-5-2. O número de esquemas táticos padrões é pequeno, mas deve se levar em conta que um esquema pode sofrer alterações. Alguns dos sistemas utilizados em Copas do Mundo são mostrados na Figura 2 (VENDITE; MORAES, 2006). Existem também outros sistemas utilizados de acordo com Wilson (2013).

Entender a técnica utilizada por adversários é considerada de extrema importância para se adquirir respostas positivas dentro de um jogo, conseguir observar o esquema usado pelo oponente em partidas anteriores e até mesmo em um tempo anterior da partida atual pode ajudar na definição da tática de um time ou na alteração da mesma (CARVALHO; SCAGLIA; COSTA, 2013).

2.2 Técnicas de Processamento de Imagem

As técnicas de processamento de imagens são utilizadas para transformar uma imagem de forma sucessiva com o objetivo de extrair mais facilmente as informações nela presente (ALBUQUERQUE, 2000).

2.2.1 HSV

HSV é a abreviatura utilizada para o sistema de cores formado pelas componentes matiz (*Hue*), saturação (*saturation*) e valor (*value*). Matiz, ou tonalidade, corresponde

Figura 2 – Esquemas táticos utilizados na história do futebol.



Fonte: Adaptado de Ribeiro (2013)

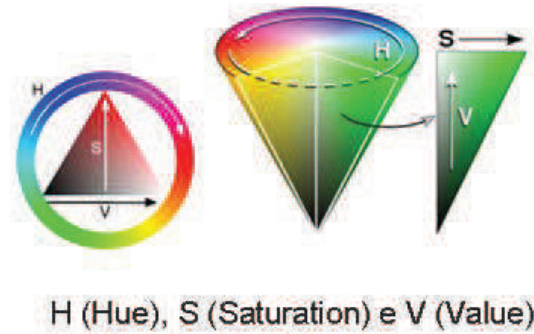
as cores, indo do vermelho ao azul em uma volta de 360°. Saturação, ou pureza, indica a quantidade de cinza presente na imagem. Valor, ou brilho, define, como o nome indica, o grau de brilho da imagem. Seus valores variam de 0 à 100%.

A Figura 3 mostra como estão organizadas as cores em HSV.

2.2.2 Thresholding

Thresholding é uma técnica de segmentação utilizada para extrair pixels de uma imagem que representam certos objetos. Esses objetos normalmente são identificados por características semelhantes, uma dessas características é a intensidade. A saída desse procedimento é uma imagem binária que separa os *pixels* do objeto de interesse dos demais

Figura 3 – Organização dos canais em HSV



Fonte: Meneguette (2009)

pixels da imagem. (MORSE; UNIVERSITY, 2000). A Equação 2.1 define a criação da imagem.

$$g(x, y) = \begin{cases} 1 & \text{SE } f(x, y) \geq T \\ 0 & \end{cases} \quad (2.1)$$

onde T representa um limiar especificado conforme os atributos do objeto procurado.

2.2.3 Filtros Lineares

Um filtro é chamado de linear quando sua aplicação resulta na substituição de um pixel pela soma ponderada de seus *pixels* vizinhos (LAGANIÈRE, 2011). Matematicamente, essa operação é chamada de correlação, e é definida pela Equação 2.2 (SZELISKI, 2011).

$$g(i, j) = \sum_{k,l} f(i+k, j+l)h(k, l) \quad (2.2)$$

onde f é a matriz da imagem e h é a máscara ou kernel, as entradas nessa máscara são comumente chamadas de coeficientes do filtro (SZELISKI, 2011). Existem diversos tipos de filtros lineares, com funções diferentes, neste trabalho foi utilizado o filtro linear da Gaussiana.

2.2.3.1 Suavização Gaussiana

O filtro da Gaussiana é amplamente utilizado para diminuir ruídos em imagens. O filtro é aplicado ao se passar todos pontos de um vetor de entrada por um *kernel* Gaussiano e somando esses pontos para produzir a saída (SZELISKI, 2011).

Na Gaussiana o ponto central tem maior peso, já o peso dos seus vizinhos vai diminuindo de forma proporcional a distância desse ponto (OPENCV.ORG, 2014b). Um

exemplo de máscara desse filtro é demonstrado na Figura 4. A Figura 5a mostra o *kernel* Gaussiano em filtros 1D e a Figura 5b para filtros 2D.

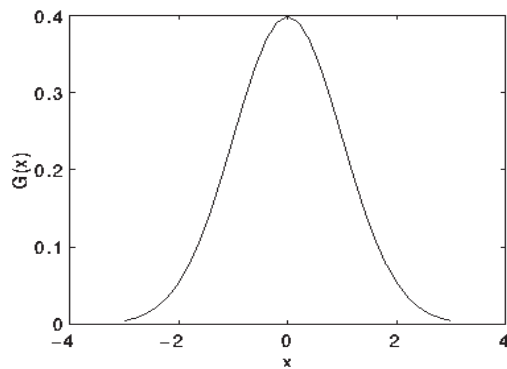
Figura 4 – Exemplo de uma máscara do filtro Gaussiano onde a máscara possui um tamanho de 5x5, sendo 273 a soma de todos os valores na mesma

$$\frac{1}{273}$$

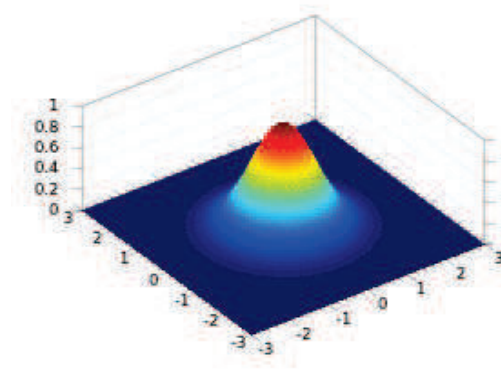
1	4	7	4	1
4	16	26	16	4
7	26	41	26	7
4	16	26	16	4
1	4	7	4	1

Fonte: Fisher et al. (2003)

Figura 5 – *Kernels* do Filtro Gaussiano em 1D e 2D



(a) Representação do *kernel* 1D



(b) Representação do *kernel* 2D

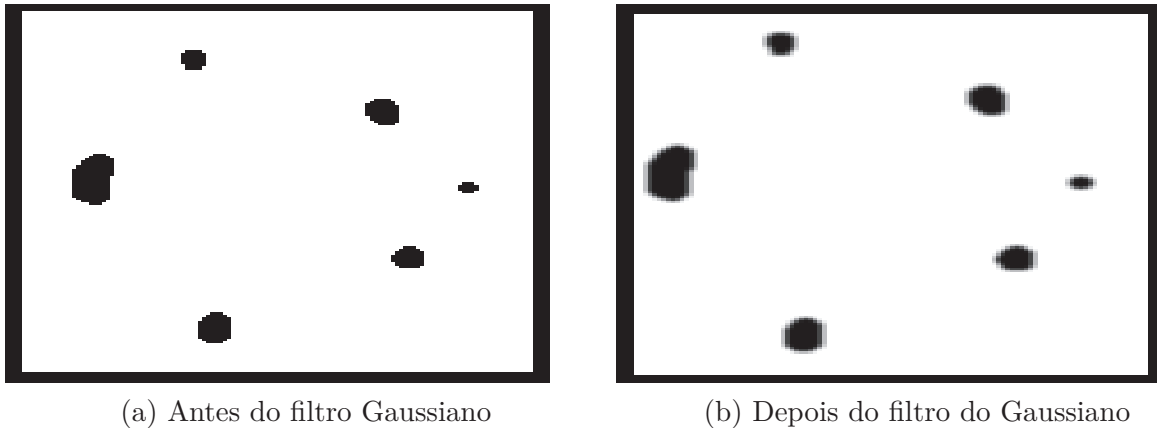
Fonte: Adaptado de Baek e E.Jacobs (2011)

Na metodologia deste trabalho a Gaussiana foi utilizada para diminuir os ruídos na obtenção da máscara de detecção de jogadores. A Figura 6b mostra a utilização desta técnica no *frame* da Figura 6a.

2.2.4 Median Blur

O *Median Blur* é uma técnica que tem como objetivo desfocar a imagem com o uso de um filtro de média. O filtro de média utiliza uma matriz de tamanho $ksize \times ksize$,

Figura 6 – Exemplo - Antes e depois da aplicação do filtro da Gaussiana. Na Figura 6b pode se observar que as formas da Figura 6a estão definidas de forma mais clara.

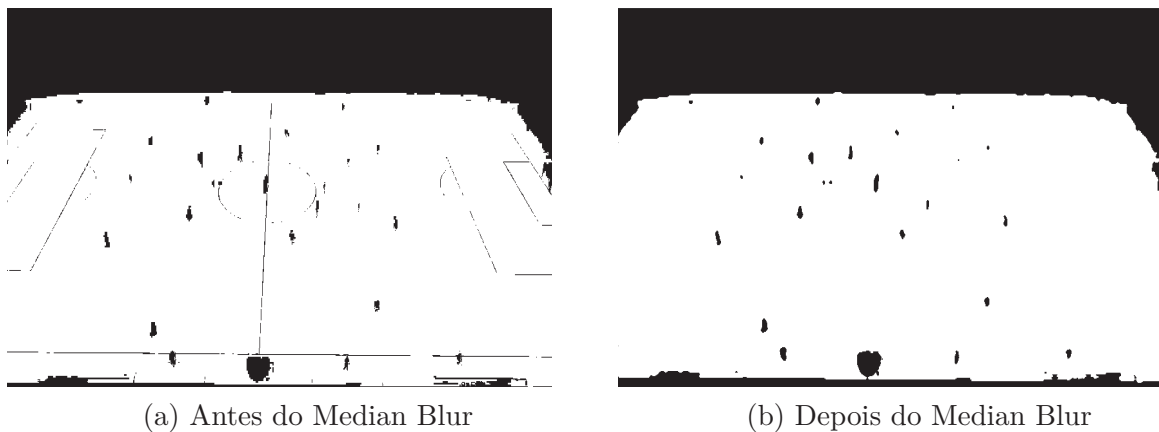


Fonte: Acervo do autor.

onde `ksize` deve ser um número ímpar (OPENCV.ORG, 2014a). Essa técnica foi usada na extração da máscara de detecção de jogadores, para a retirada das linhas do gramado.

O uso desta técnica no trabalho é demonstrado na Figura 7.

Figura 7 – Exemplo - Antes e depois da utilização do *Median Blur*. Na Figura 7b podemos observar que as linhas de campo da Figura 7a foram retiradas.



Fonte: Acervo do autor.

2.2.5 Operação Morfológica de Fechamento

Operações morfológicas são utilizadas no processamento de imagem executando operações não lineares (DAWSON-HOWE, 2014). Fazendo uso de um elemento estruturante¹, essas operações executam teste lógicos em todas as posições possíveis da imagem entre esse elemento e uma porção da imagem correspondente, o resultado é então armazenado em uma

¹ Elementos estruturantes são pequenos conjuntos de pontos definidos em volta de sua origem

imagem de saída (DAWSON-HOWE, 2014). Dentre as principais operações morfológicas estão a dilatação, erosão e a operação de fechamento.

2.2.5.1 Dilatação

Dilatação é uma operação morfológica que expande os números de *pixels* de um objeto, normalmente em todas as direções de forma simultânea. Realizando assim o preenchimento de pequenos vazios e a junção entre grandes grupos de pontos que estivessem a uma pequena distância entre si (DAWSON-HOWE, 2014). A Equação 2.3 apresenta a função de dilatação morfológica:

$$A \oplus B = \{z | (\hat{B}) \cap A \neq \emptyset\} \quad (2.3)$$

onde A é a imagem e B é o elemento estruturante.

2.2.5.2 Erosão

Erosão é uma técnica que realiza a redução de um objeto, realizando a remoção de *pixels* de suas bordas. Executando assim a melhor separação de objetos e a retirada de pequenos ruídos (DAWSON-HOWE, 2014). A função da erosão está definida na Equação 2.4.

$$A \ominus B = \{x | (B)_x \subseteq A\} \quad (2.4)$$

onde A é a imagem e B é o elemento estruturante.

2.2.5.3 Operação de Fechamento

Uma operação de fechamento consiste na execução de uma dilatação seguida de uma erosão com o mesmo elemento estruturante. O fechamento junta objetos que estão próximos e preenche vazios em objetos. Esse método tende a causar distorção em objetos. A função de fechamento é demonstrada na Equação 2.5.

$$A \bullet B = (A \oplus B) \ominus B \quad (2.5)$$

onde B é o elemento estruturante e A é a imagem.

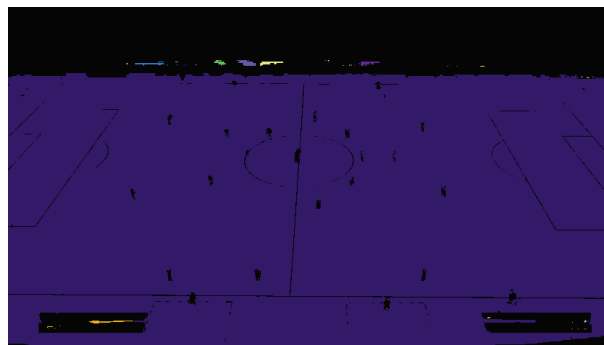
2.2.6 Componentes Conectados

Rotulação de Componentes conectados é um algoritmo aplicado na teoria dos grafos, onde os grupos de componentes conectados são separados de acordo com uma

¹ DAWSON-HOWE, K. *A Practical Introduction to Computer Vision with OpenCV*. 2014.

heurística dada (SALEM, 2014). Em visão computacional é utilizado para detectar regiões em imagens binárias. Ele funciona da seguinte forma: Considerando cada pixel como um vértice, e arestas como vizinhos que estejam conectados. Um algoritmo percorre a imagem, rotulando vértices baseados na sua conexão com seus vizinhos (SAMET; TAMMINEN, 1988). Neste trabalho componentes conectados foram utilizados para realizar a obtenção da máscara de campo. Um exemplo do uso está na Figura 8, nela estão presentes os componentes encontrados pelo algoritmo separados por cores, onde o componente de cor roxa representa o campo.

Figura 8 – Exemplo de uso do algoritmo de Componentes Conectados com os componentes separados por cores



Fonte: Acervo do autor.

2.2.7 Detecção de Bordas Canny

O detector de bordas Canny realiza a combinação da primeira e segunda derivada da detecção de borda para obter a magnitude e orientação dessa borda (DAWSON-HOWE, 2014). Esse filtro foi criado para otimizar os seguintes fatores: detecção, localização e resposta.

A melhora na detecção tem como objetivo diminuir o número de arestas perdidas, a localização busca reduzir a distância entre a aresta encontrada e sua real localização e resposta busca um menor número de resultados duplicadas para uma mesma aresta.

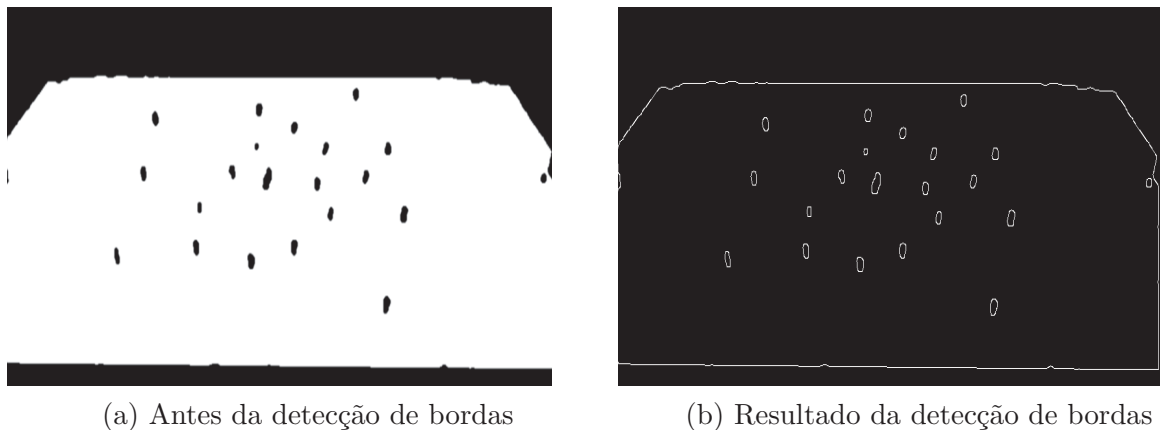
As etapas presentes no algoritmo de Canny (CONCI, 2007):

1. Convolui-se a imagem com uma máscara Gaussiana. Realizando assim o desfoque da imagem, e dessa forma reduzindo o ruído da mesma.
2. Calcula-se o gradiente de intensidade da imagem, já que a borda da imagem pode estar em vários sentidos. Logo, 4 máscaras são aplicadas: horizontal, vertical e duas diagonais.

3. Os resultados são então armazenados e para cada pixel, marca-se a maior resposta de gradiente para o mesmo, e qual máscara produziu essa borda. Criando assim dois mapas. Um de gradiente de intensidade e outro para o sentido.
4. Assumindo-se que os gradientes mais elevados possuem maior probabilidade de serem bordas, a detecção de bordas é então realizada. Porém, o algoritmo de Canny propõe o uso de dois limites, inferior e superior, já que não há um valor absoluto que determine quando uma intensidade é uma borda.

Um exemplo do uso do Canny neste trabalho está na Figura 9, nele todas as bordas dos objetos presentes na Figura 9a foram detectadas e exibidas na Figura 9b. O Canny foi utilizado na máscara de detecção de jogadores.

Figura 9 – Exemplo - Detecção de bordas com Canny



Fonte: Acervo do autor.

2.2.8 A transformada de Hough

A transformada de Hough foi desenvolvida para detectar formas que podem ser parametrizadas de forma simples, como linhas, círculos e elipses, em imagens binárias. Essa técnica é aplicada após a imagem sofrer um pré-processamento, normalmente a detecção de bordas (JAMUNDA, 2000).

Esse procedimento consiste em realizar o mapeamento de um pixel da imagem em um espaço de parâmetros (Espaço de Hough). O método original de Hough utilizava a parametrização Inclinação-Intersecção para mapear um ponto na imagem para o plano de parâmetros empregando a Equação 2.6 (MACEDO, 2005).

$$y = ax + b \quad (2.6)$$

Para minimizar o custo da parametrização, uma forma alternativa da equação é utilizada. Essa forma é mostrada na Equação 2.7.

$$s = i\cos\theta + j\sin\theta \quad (2.7)$$

onde s é a distância ortogonal da linha para a origem e θ é o ângulo dessa normal com o eixo horizontal (DAWSON-HOWE, 2014).

Com a utilização desses novos parâmetros o problema de detecção de pontos colineares pode ser convertido para encontrar curvas concorrentes. Dessa forma para se calcular a transformada de Hough se utiliza uma matriz que inicialmente contém apenas valores nulos, as dimensões da matriz dependem dos intervalos previamente fixados pelos eixos s e θ do espaço de parâmetros. Para cada ângulo encontrado do intervalo $[0, \pi]$ e para cada ponto (x,y) da imagem de entrada, um novo valor de s é encontrado, a partir da equação, após o cálculo, a célula da matriz acumuladora na posição (θ, s) é acrescentada em 1 (DAWSON-HOWE, 2014). Analisando um *pixel* na imagem, procura-se por todas as curvas que passam por este ponto, e para cada curva a posição do acumulador correspondente aos parâmetros desta curva é incrementado. Assim o máximo do acumulador corresponderá ao objeto procurado (SILVA, 2010).

A transformada foi utilizada neste trabalho para encontrar o meio-campo, de forma a se obter o modelo dos times, já que os mesmos iniciam o jogo divididos pela linha de centro. Essa linha é vertical e está localizada na parte central da imagem.

2.3 Técnicas de Aprendizagem de Máquina

Aprendizado de máquina é a capacidade de desenvolver em computadores a habilidade de aprender sem precisar de uma programação explícita. Existem muitos tipos de técnicas nesta área, neste trabalho será utilizado um classificador. Um sistema que recebe características discretas ou contínuas e retorna um único valor discreto, chamado de classe (DOMINGOS, 2012).

2.3.1 K-Nearest-Neighbor

K-Nearest-Neighbor (KNN) é um classificador de objetos que fundamenta-se em uma base de dados previamente treinada (HARRINGTON, 2012). Esse treinamento é realizado com a atribuição de rótulos (classes) para um conjunto de objetos que possuem determinadas características. Para realizar a classificação, sempre que o KNN receber um novo dado, ele o categoriza com uma das classes configuradas. Essa classificação se baseia na distância entre os dados do novo objeto e os dados na base de treino. A classe que estiver mais presente entre as k menores distâncias encontradas, será a classe do novo objeto (HARRINGTON, 2012).

A principal métrica utilizada pelo KNN para medir esse valor é a distância Euclidiana, mostrada na Equação 2.8, que calcula a menor distância entre dois pontos. Porém outras métricas podem ser utilizadas, como a distância de Manhattan, Equação 2.9, que calcula a distância entre dois pontos obtendo a soma das distâncias de suas coordenadas (MATHWORKS, 2016).

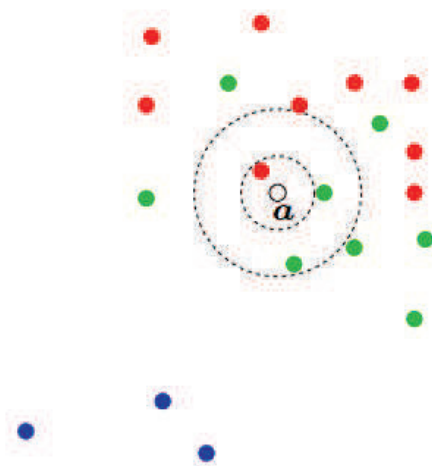
$$d(v, a) = \sqrt{\sum_{j=1}^n (v_j - a_j)^2} \quad (2.8)$$

$$d(v, a) = \sum_{j=1}^n |(v_j - a_j)| \quad (2.9)$$

onde $v = (v_1, v_2 \dots v_n)$ e $a = (a_1, a_2 \dots a_n)$ são dois pontos no \mathbb{R}^n .

Para realizar a classificação de objetos baseados nos dados de treino se é escolhido um k . Esse k consiste no número de vizinhos que irão influenciar a classificação, esse k é em geral um número ímpar, para evitar empates, mas isso só é uma realidade para casos onde somente 2 rótulos estão presentes (SONG et al., 2007). A Figura 10 mostra um conjunto de pontos vermelhos, azuis e verdes e um ponto a . A classificação de a é diferente dependendo do número k definido. Para $k = 1$, o a seria classificado como vermelho, para $k = 3$, a seria classificado como verde.

Figura 10 – Exemplo - KNN ²



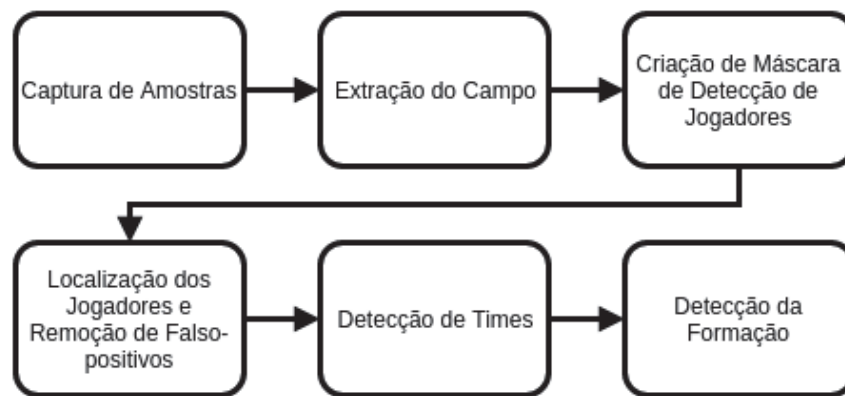
Fonte: Thirumuruganathan (2010)

3 Metodologia

Este capítulo apresenta a metodologia proposta para a detecção de esquemas táticos de times de futebol durante uma partida.

A proposta se fundamenta no uso de conhecimentos de visão computacional, processamento de imagem e análise estatística para realizar a obtenção de formação tática, propiciando assim uma ferramenta de pesquisa para profissionais da área, bem como aficionados pelo esporte. A Figura 11 exemplifica o funcionamento geral da metodologia. As etapas são descritas na seções seguintes.

Figura 11 – Visão geral da metodologia desenvolvida.



Fonte: Acervo do autor.

3.1 Captura de Amostras

As amostras usadas neste trabalho foram extraídas do jogo de videogame FIFA *Soccer* - 2014, elas foram obtidas pela modificação da câmera de jogo, para aumentar a sua altura e permitir uma melhor visualização do gramado. A altura e dados da câmera estão informados na Tabela 1. Utilizar o jogo como base de dados para a execução da metodologia foi necessário pois as imagens geradas por emissoras de televisão são voltadas para a melhor visualização de jogadas específicas. Para realizar a obtenção de amostras em jogos reais, uma câmera fixa com um ângulo de visão amplo, de modo a exibir todos os jogadores, e posicionada de forma a ter a linha central exibida na área central da imagem deve ser utilizada.

Tabela 1 – Parâmetros de jogo modificados para obtenção de uma melhor visão do gramado

Parâmetros	Valores
CAMERA_ZOOM_MODIFIER_COOP	0.09
CAMERA_HEIGHT_MODIFIER_COOP	300.1

Fonte: Acervo do autor.

3.2 Extração do Campo

A obtenção do gramado é a primeira parte da resolução do problema, pois nele estão contidos jogadores, bem como juízes, técnicos e a bola. Em (CHOI et al., 1997) o campo é caracterizado como o objeto de maior extensão no vídeo e que mantém uma cor verde dominante. E sua extração é realizada com a conversão dos *frames* para HSV e a retirada de *pixels* em que os limites inferiores e superiores de saturação, citados por (KHATOONABADI; RAHMATI, 2009) e mostrados na Tabela 2, sejam ultrapassados, gerando assim uma imagem binária, mostrada na Figura 12a.

Tabela 2 – Limites de Saturação

Limites	Valores
Limite Inferior	54
Limite Superior	90

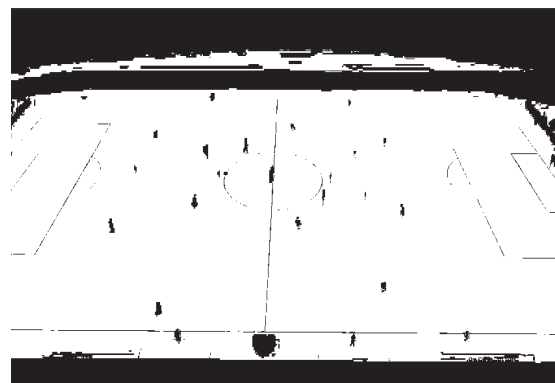
Fonte: (KHATOONABADI; RAHMATI, 2009)

Essa abordagem porém pode apresentar problemas em jogos onde a arquibancadas também apresenta cores dominantes de verde ou onde existam regiões de gramados não adjacentes ao campo, como demonstra a Figura 12b.

Figura 12 – Exemplo - extração do campo em jogos diferentes



(a) Retirada do verde

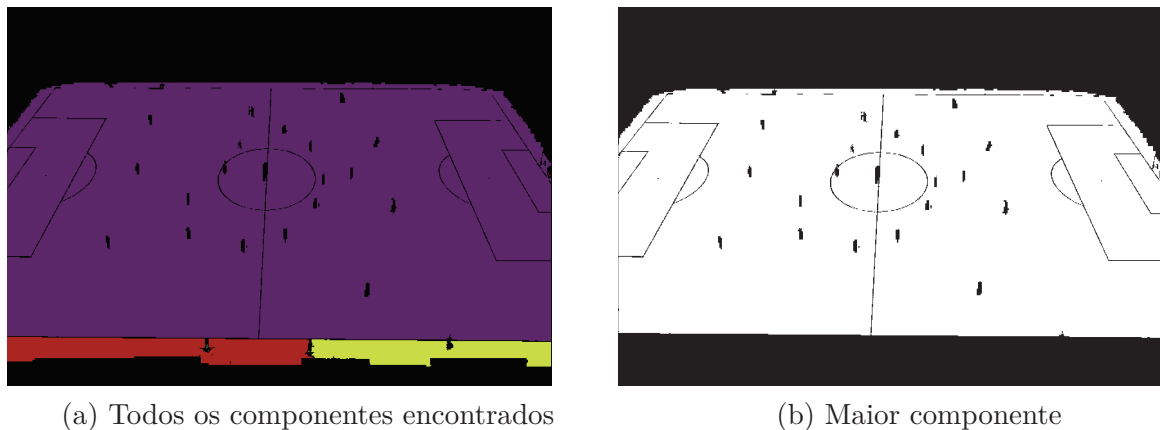


(b) Retirada do verde com áreas verdes não adjacentes

Fonte: Acervo do autor.

O problema apresentado foi resolvido com o uso do algoritmo *Connected Components* que tem como objetivo encontrar *pixels* adjacentes que possuem o mesmo valor, o retorno desse algoritmo nos trás o número de componentes conectados encontrados e suas localizações. Assim, é criada uma nova imagem que receberá apenas os *pixels* presentes no componente de maior área, já que entende-se que o campo ocupa a maior parte do vídeo (CHOI et al., 1997). A Figura 13b mostra o maior componente encontrado na Figura 13a.

Figura 13 – Uso do *Connected Components* para obtenção da máscara de campo



Fonte: Acervo do autor.

Como pode ser observado na Figura 13b, a imagem ainda possui diversas lacunas, derivadas dos jogadores, linhas ou outros elementos que estavam no gramado. A máscara de campo é então encontrada ao realizarmos o preenchimento desses espaços, (CHOI et al., 1997).

A máscara de campo deve ser uma imagem semelhante a Figura 14.

Figura 14 – Exemplo de uma máscara de campo



Fonte: Acervo do autor.

3.3 Criação de Máscara de Detecção de Jogadores

A máscara de detecção de jogadores deve ser obtida para definir a área dos jogadores na imagem. Essa máscara não é exclusiva, pois nela podem estar incluídos quaisquer objeto que estejam na área de gramado detectada pela máscara de campo. A sua obtenção é realizada a partir da máscara do campo antes do preenchimento (Figura 13), após utilizar sobre ela as seguintes operações: filtros lineares, para redução de ruídos e operações morfológicas de fechamento, para obter melhor separação entre jogadores. O conjunto dessas operações também é responsável pela eliminação das linhas do campo na imagem, esse procedimento foi realizado por [Khatoonabadi e Rahmati \(2009\)](#), com a utilização de um filtro, mostrado na Equação 3.1 na imagem em escala de cinza do *frame*. Exemplos de resultados desse procedimento são mostrados na Figura 15.

$$\begin{pmatrix} -2 & 1 & -2 \\ 1 & 4 & 1 \\ -2 & 1 & -2 \end{pmatrix} \quad (3.1)$$

Figura 15 – Exemplo - Máscara de detecção de jogadores em jogos diferentes



Fonte: Acervo do autor.

3.4 Localização dos Jogadores e Remoção de Falso-positivos

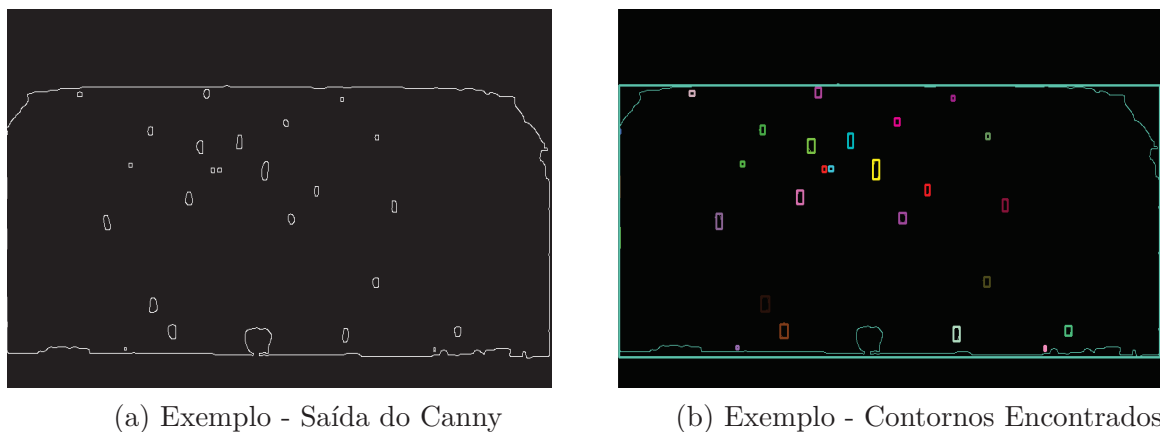
Para encontrar os jogadores, localizamos primeiro as bordas dos objetos dentro da máscara de detecção. Isso é feito com a utilização do detector de bordas Canny aplicado sobre ela, Figura 16a.

Para obtermos as bordas utilizamos o algoritmo criado por [Suzuki e Abe \(1985\)](#) que nos retorna uma lista de contornos, onde cada elemento da lista é um vetor de pontos que formam cada contorno. Usando essa lista, o menor retângulo que envolva cada contorno é

obtido com o uso do algoritmo Ramer–Douglas–Peucker. A Figura 16b mostra os retângulos encontrados para a Figura 16a.

Como discutido na seção anterior, a máscara de detecção de jogadores captura bem mais que apenas os jogadores, após encontrarmos todos os elementos dentro do campo, é necessário separá-los, para que se possa obter apenas os atletas dos times que estejam em campo.

Figura 16 – Exemplo - Utilização do Canny no encontro de contornos



Fonte: Acervo do autor.

O tamanho esperado de um jogador com a câmera na distância configurada na seção de obtenção das imagens é de 40 *pixels* de altura e 20 de largura. Elementos que possuem valor de área desproporcional a esse tamanho são retirados. Para remover falso-positivos é feita a média truncada da área dos retângulos restantes. Essa média é calculada com a retirada dos extremos (maior e menor retângulo) e em seguida, computando-se a média simples. Qualquer área de um retângulo que é maior que a média é colocado em outra lista, já que poderemos utilizar esses retângulo posteriormente. Dessa forma, apenas pessoas no gramado serão detectadas pelo algoritmo.

Com essa aplicação, a detecção terá obtido apenas elementos no gramado que forem similar a um jogador. Como em alguns campos a área verde não é delimitada pelas 4 linhas de jogo, objetos e pessoas que estejam presentes nesta área não serão excluídas pelo algoritmo de extração de campo. A Figura 17 mostra um exemplo desse tipo de gramado, onde é possível observar que o campo se estende além das 4 linhas de jogo. Para solucionar esse problema a identificação de falso-positivos detectados nessa área é realizada.

A identificação desses objetos é feita utilizando-se dos retângulos que possuem área acima da média e que foram alocados em uma lista separada. Dessa lista, encontramos o maior retângulo que tenha uma posição y central, posição que é ocupada pelo gol. Essa posição é dada quando realizamos a divisão da tela de forma horizontal em 4 partes iguais, as duas partes que se localizam no meio, são consideradas centrais. A Figura 18 mostra

Figura 17 – Exemplo - Campo com áreas verdes adjacentes. É possível observar que a área verde continua mesmo com o fim do campo de jogo.



Fonte: Acervo do autor.

essa divisão e a localização das partes centrais.

Figura 18 – Exemplo - Linhas utilizadas para dividir o campo em 4 partes. A parte central são as áreas que se localizam entre as linhas amarelas.



Fonte: Acervo do autor.

Após o encontro desses retângulos, que são chamados de gols, todos os objetos detectados antes do gol esquerdo e após o gol direito são removidos.

Como no gramado podem se encontrar técnicos e auxiliares as linhas de delimitação lateral são localizadas com o uso da Transformada de Hough, um exemplo dessas linhas está na Figura 19a. Isso é feito encontrando-se os segmentos de reta que constam na primeira e última porção da divisão horizontal do *frame* que foi mostrada na Figura 18. Na primeira faixa é encontrada a linha de menor altura, essa linha corresponde a lateral superior do campo. Na última porção se encontra a linha correspondente a lateral inferior. As linhas encontradas são expandidas de forma que elas possuam a mesma largura da imagem. A Figura 19b mostra a expansão das linhas da Figura 19a. Pessoas que estejam acima da linha superior e abaixo da linha inferior são eliminadas.

Como a localização de pessoas que não sejam jogadores pode afetar o resultado de

Figura 19 – Exemplo - As linhas encontradas pela Transformadas de Hough e suas formas estendidas



(a) Exemplo - Linhas laterais encontradas



(b) Exemplo - Linhas laterais estendidas

Fonte: Acervo do autor.

forma maior do que perder jogadores na detecção, se alguma das linhas correspondentes a encontradas, os limites da divisão da tela (com um pequeno recuo) serão utilizadas como referência para eliminação dessas pessoas. Exemplo dessas linhas é mostrado na Figura 20.

Figura 20 – Exemplo - Linhas projetadas com o uso da divisão horizontal da tela em 4 partes.



Fonte: Acervo do autor.

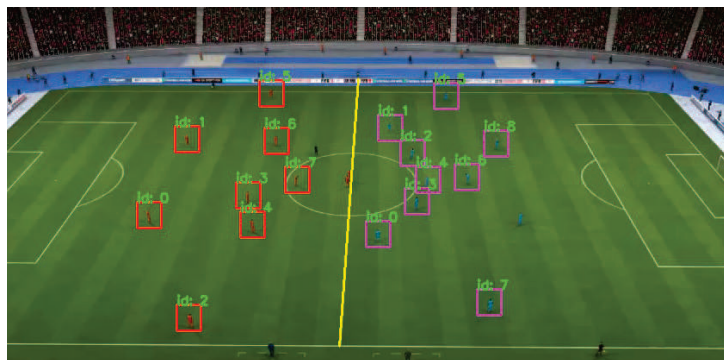
3.5 Detecção de Times

A detecção de times foi realizada com a execução do algoritmo KNN. O algoritmo utiliza uma base de dados treinada para classificar objetos em determinados rótulos. Para treinar a base de dados foram utilizados modelos de cada time criando assim duas classe, uma para o primeiro time e outra para o segundo. A criação do modelo foi realizada localizando-se os atletas no primeiro *frame* de jogo, onde a maioria dos jogadores está incluído, de forma a ter a melhor base possível.

3.5.1 Criação dos Modelos de Times

A obtenção desses modelos é feita a partir da localização da linha central. Para adquirir essa linha, a transformada Hough foi utilizada na imagem em escala de cinza do primeiro *frame*. Aproveitando-se do fato que um jogo de futebol tem no seu início seus times postados em lados opostos no campo e que a exibição de um jogo inicia com foco da linha central, todos os jogadores encontrado à esquerda dessa linha são alocados no time 1 e os jogadores à direita, no time 2. A Figura 21 mostra essa divisão.

Figura 21 – Exemplo - Separação dos times com base na linha central.



Fonte: Acervo do autor.

Com os jogadores alocados em suas respectivas equipes, as características das mesmas foram obtidas com a divisão de cada retângulo correspondente a um jogador pela metade, de forma a se criar 2 partes, a superior e a inferior. As Figuras 22b e 22c mostram as partes, superior e inferior, da Figura 22a. Em seguida realiza-se o cálculo do histograma dessas partes para os três canais presentes em imagens com o formato de cores HSV: uma para a matiz, a saturação e o valor. No final desse procedimento, obtemos 6 histogramas para cada jogador. Esse conjunto é então classificado com o rótulo correspondente à equipe do jogador. Finalizando assim, o treinamento da base de dados do KNN com o uso do primeiro *frame* de jogo.

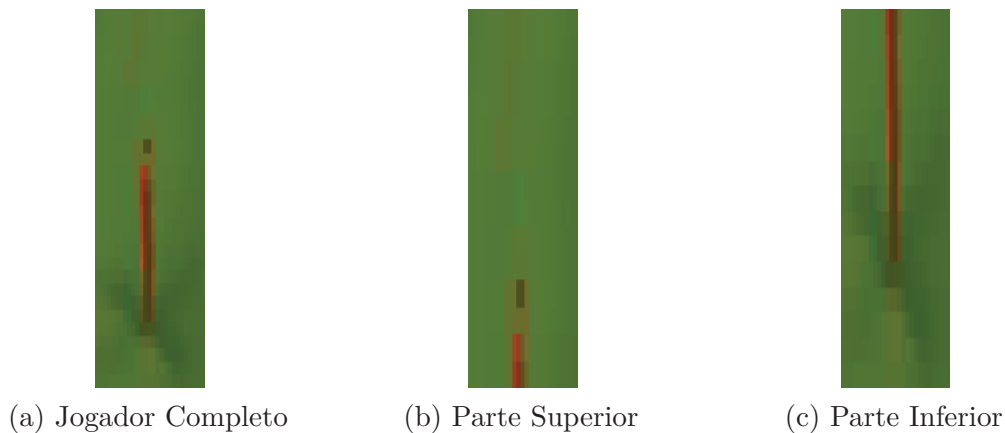
3.5.2 A classificação do KNN

A cada *frame*, o resultado da detecção dos jogadores é adicionado ao KNN, que classifica-os com base em seus modelos treinados. Dessa forma, todos os atletas que forem detectados serão atribuídos a um dos times.

3.5.3 Matriz de Classificação

Para melhorar a classificação do KNN, foi criada uma matriz de classificação. Essa matriz possui um número de linhas correspondente a quantidade de jogadores nesse *frame*, e duas colunas que contém, respectivamente, a posição central do retângulo de cada jogador

Figura 22 – Exemplo - Divisão do jogador em duas partes. A Figura 22a mostra o jogador completo e as Figuras 22b e 22c mostram, respectivamente, a parte superior e inferior do jogador.



Fonte: Acervo do autor.

Tabela 3 – Matriz de Classificação formada a partir da Figura 21

Ponto central do retângulo (x,y)	Time
(171, 249)	1
(215, 161)	1
(286, 226)	1
(291, 260)	1
(437, 271)	2
(451, 148)	2
(477, 177)	2
(482, 233)	2

Fonte: Acervo do autor.

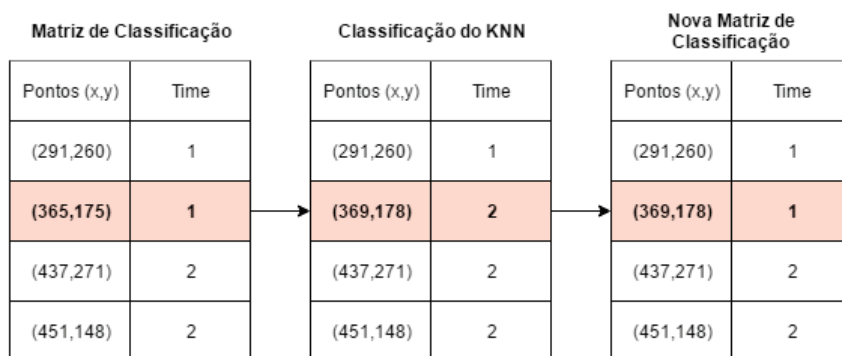
e o time a qual ele foi atribuído. Antes de adicionar os jogadores nesta matriz, os atletas são organizados com base na coordenada x do ponto central do seu retângulo, em ordem crescente.

A Tabela 3 mostra um exemplo de matriz construída para o primeiro *frame* (Figura 21) de um jogo. Nesse *frame* os times são exibidos segregados por cores e com a indicação do índice de cada jogador sobre o seu retângulo.

Em todos os outros *frames* do vídeo, a matriz será comparada com a classificação fornecida pelo KNN e sempre que a diferença entre a posição de um jogador encontrada no *frame* e um jogador presente na matriz for menor que 15 *pixels*, eles são considerados o mesmo jogador, e a classificação do KNN é substituída pela classificação da matriz e a posição do jogador na matriz é atualizada com a posição do jogador no novo *frame*. Dessa forma a matriz estará com as posições atualizada, minimizando sua taxa de erro.

O diagrama da Figura 23 mostra como funciona esse sistema. O elemento em negrito é considerado o mesmo elemento na tabela da matriz de classificação e na tabela de classificação do KNN pois eles estão a uma diferença menor que 15 *pixels*. Logo, o time escolhido é o determinado pela matriz de classificação, mas a posição será a mesma da tabela do classificador KNN.

Figura 23 – Exemplo - Funcionamento da matriz de classificação.



Fonte: Acervo do autor.

3.6 Detecção da Formação

A detecção de formação é dividida em 4 etapas: a leitura das formações, a criação das linhas de campo, o encontro dos separadores e a busca pelas formações possíveis.

A base de formações que o algoritmo tentará encontrar nos jogos é mostrada na Tabela 4. Elas foram armazenadas para que a comparação com formação encontrada possa ser realizada.

Tabela 4 – Formações a serem encontradas

Formação	Defensores	Meio-campistas	Atacantes
5-3-2	5	3	2
5-4-1	5	4	1
4-5-1	4	5	1
4-4-2	4	4	2
4-3-3	4	3	3
4-2-4	4	2	4
3-5-2	3	5	2
3-4-3	3	4	3

Fonte: Acervo do autor.

Para encontrar a formação utilizada por uma equipe, é necessário conhecer a distância em que os jogadores de um mesmo time estão um dos outros. Devido a

perspectiva da filmagem, a simples distância entre os retângulos dos jogadores não resulta no espaçamento real entre eles. Para solucionar este problema, utiliza-se como referência a linha central, que será replicada entre o primeiro e o último jogador de um time. A Figura 24 demonstra essa replicação.

Figura 24 – Exemplo - Criação de replicações da linha central entre o primeiro e último jogador de um time. As linhas em lilás são chamadas de separadoras.



Fonte: Acervo do autor.

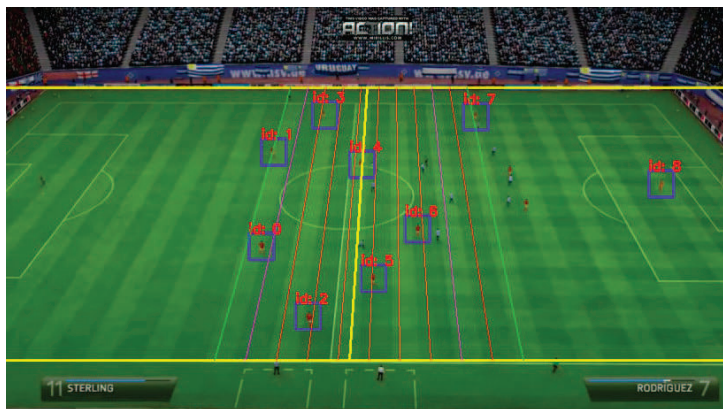
A replicação é realizada da seguinte maneira: com a lista de jogadores de um time ordenada pela coordenada x , é feito o cálculo da distância horizontal entre o primeiro e o último jogador, de forma que se possa obter o número de linhas que podem ser traçadas entre eles. Considerando que a distância entre uma linha e outra foi configurada como 30 *pixels* (tamanho relativo ao tamanho do *bounding box* do jogador) obtemos então a diferença entre a linha central (que é armazenada, ou seja, ela não precisa aparecer em todos os *frames*) e o primeiro jogador para descobrir o local de início das linhas.

Depois que as linhas foram criadas, armazena-se qual linha se encontra mais perto de cada jogador. Se o primeiro ou último jogador estiverem a uma distância superior a 4 linhas do seu vizinho mais próximo, esse jogador é descartado. Isso é feito para evitar que os goleiros possam aparecer nas formações. Um exemplo disso está na Figura 25. As distâncias calculadas serão usadas para a atribuição de jogadores a uma das partes da formação: ataque, meio-campo e defesa. As linhas de separação entre essas partes são chamadas de linhas separadoras.

3.6.1 Encontro das Linhas Separadoras

O descobrimento das linhas separadoras é feito com a definição das linhas que marcam o começo do ataque (separadora 0) o fim do ataque (separadora 1) e início da defesa (separadora 2). Para o time dois esses separadores são invertidos. Todos os jogadores entre a separadora 1 e a separadora 2 são considerados meio-campistas. A Figura 24 mostra um exemplo dessas linhas em um *frame*.

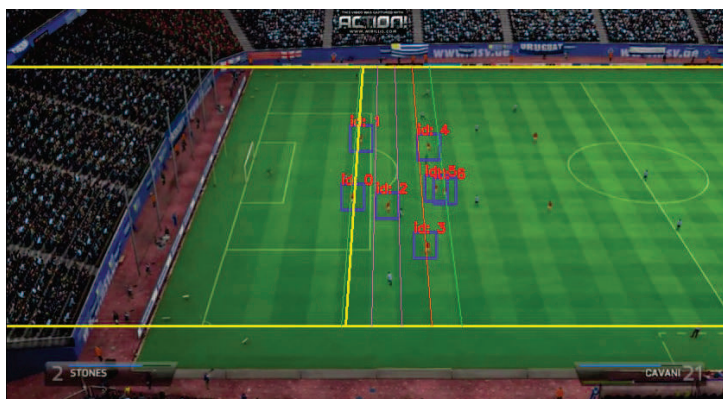
Figura 25 – Exemplo - Uso das Linhas de Campo para Retirada dos Goleiros. O jogador de número 8 é retirado da contagem devido a distância dele para o jogador mais próximo (número 7)



Fonte: Acervo do autor.

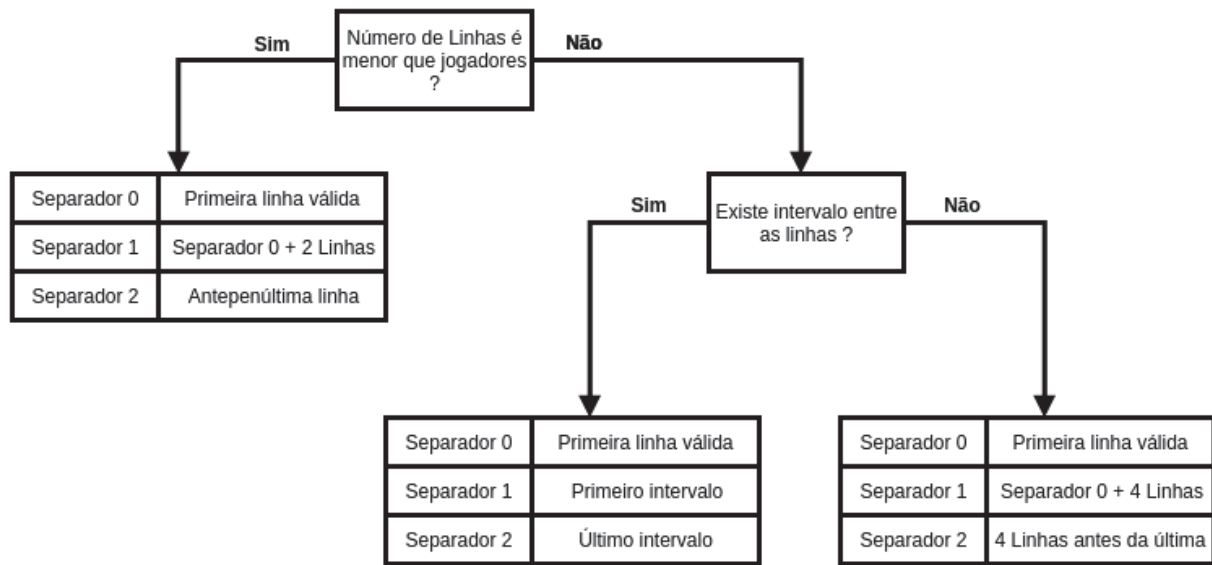
Existem 3 formas para definir a separação entre as linhas: a primeira ocorre caso o número de linhas seja menor que o número de jogadores, a segunda é aplicada quando o número de linhas é maior que o de jogadores e existe algum intervalo de linhas entre atletas, a terceira é selecionada sempre que o número de linha também é maior que o de jogadores, mas não existe intervalo entre essas linhas. A Figura 27 mostra os valores usados para encontrar esses separadores, caso exista um intervalo seguido de outro, o último intervalo é utilizado. Se a diferença entre o primeiro jogador e o último for menor que 5 linhas, a informação do *frame* é descartada, pois se considera um momento de jogo de indefinição, um exemplo desse momento é mostrado na Figura 26. Pode-se observar que o número de jogadores detectados é pequeno e eles estão muito próximos, tornando a detecção de formação neste *frame* inviável.

Figura 26 – Exemplo - *Frame* descartado devido ao pequeno número de linhas. As linhas em amarelo são: linha superior, linha central e linha inferior.



Fonte: Acervo do autor.

Figura 27 – Fluxograma de seleção das linhas separadoras



Fonte: Acervo do autor.

3.6.2 Busca pelas Formações

A busca de formação é realizada após a análise de um conjunto de *frames* que representam determinado intervalo de tempo em um jogo. Ao final de cada intervalo, pode-se ter ou não o número de jogadores encontrados em cada região, isso acontece pois os *frames* que possuem menos de 6 ou mais de 10 jogadores detectados em cada time são descartados. A resposta dessa análise nos fornece a formação encontrada, mas essa formação nem sempre está completa, já que jogadores podem não ter sido encontrados na detecção ou não atribuídos ao time correto. Então, todas as formações tem um peso atribuído. Esse peso corresponde ao módulo da diferença entre o número de jogadores na formação original e na encontrada. O cálculo pode ser realizado com a aplicação da Equação 3.2. Ao final do intervalo de *frames* estabelecido, a formação que obtiver a maior soma de pesos será a resposta apresentada.

$$peso = \frac{1}{|D - dF| + |M - mF| + |A - aF|} \quad (3.2)$$

onde D, M e A correspondem a quantidade de jogadores na defesa, meio-campo e ataque das formações na base de dados e dF, mF e aF correspondem a quantidade de jogadores na defesa, meio-campo e ataque encontrada no *frame*.

Se ao final do intervalo de tempo, o peso das formações forem o mesmo, o cálculo da média de jogadores encontrados em cada parte da formação é realizado para se gerar uma nova formação. Se não houver desempate, as duas formações são informadas.

4 Resultados

Neste capítulo são apresentados resultados da metodologia proposta para a detecção de formação em jogos de futebol.

Os vídeos utilizados para teste foram extraídos do jogo de videogame FIFA *Soccer* - 2014, para obtenção de uma melhor visualização do gramado. Dez jogos foram utilizados nos testes (com equipes diferentes) e seus resultados foram computados por períodos de jogo (1º e 2º tempo), obtendo assim 20 períodos.

A análise da formação foi feita baseada em intervalos de tempo. Como no videogame cada período de partida pode ser configurado. A configuração utilizada nesse trabalho foi de 4 minutos por tempo de jogo. Gerando assim, 7200 *frames* para transmissão de cada tempo do jogo. Para representar 15 minutos de jogo foram obtidos resultados a cada 2400 *frames*. Em uma partida normal seriam 81000 *frames* (45 minutos em vídeos de 30fps) por período e 27000 *frames* a cada 15 minutos. Como o resultado é dado a cada 15 minutos, 3 formações são encontradas a cada tempo de jogo.

Os testes foram realizados de acordo com a metodologia, para que cada parte possa ser avaliada de forma separada e os desempenhos possam ser analisados de forma mais precisa.

A primeira parte dos testes verificou o desempenho da extração do campo. O teste de extração foi realizado em campos diferentes e que apresentavam diferentes condições climáticas, tendo resultado positivo em 90.9% dos casos. Os erros apresentados foram em campos onde o gramado estava desgastado e com neve acumulada. Um exemplo desse campo está na Figura 28.

Figura 28 – Exemplo - Campo onde a máscara gerada apresentou erro

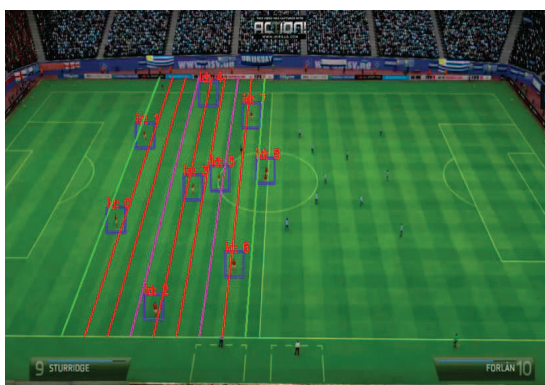


Fonte: Acervo do autor.

A próxima etapa da metodologia é a criação de máscara de Detecção de jogadores.

O teste foi realizado em conjunto com a fase de localização dos jogadores e remoção de falso-positivos. Foram testados todos os *frames* dos vídeos e foram considerados *frame* válidos aqueles onde o número de jogadores encontrados foi maior que 6 em cada time. Essa condição foi imposta porque a qualidade da informação do *frame* é mais importante que sua quantidade. Em média 23% dos *frames* de um vídeo são considerados válidos e são utilizados para a detecção de formação. A Figura 29 mostra exemplo de um *frame* considerado válido (Figura 29a) e um considerado não válido (Figura 29b).

Figura 29 – Exemplo - *Frames* válidos e não válidos. No *frame* considerado não válido o número de jogadores excede o número máximo de jogadores em um time. Um time composto apenas de jogadores de linha, sem o goleiro, possui 10 componentes.



(a) Exemplo - *Frame* válido



(b) Exemplo - *Frame* não válido

Fonte: Acervo do autor.

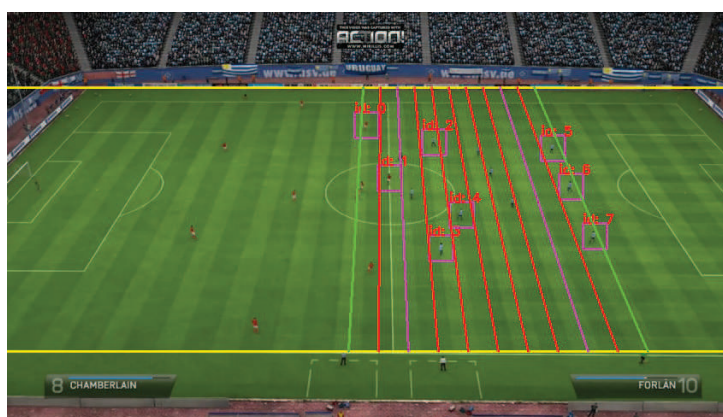
Com a fase de testes de localização de jogador concluída, os *frames* classificados como válidos foram utilizados como base de dados para a verificação da detecção de times. A acurácia apresentada foi de 61,6%. As porcentagens de acertos referentes a cada formação estão presentes na Tabela 5. Foram analisados 20 jogos, com cada jogo fornecendo a formação utilizada a cada 15 minutos de partida. No total, 60 formações foram obtidas. A formação 5 - 4 - 1 não foi apresentada em nenhum jogo. A formação 4 - 5 - 1 apareceu 4 vezes, mas obteve uma porcentagem de acerto baixa. O sistema de classificação de times contribuiu para 31 % dos erros apresentados. A Figura 30 mostra um erro de classificação de times que contribuiu para um erro no encontro da formação correta. A Figura 31 mostra um *frame* onde a formação encontrada corresponde a formação presente na imagem. A Tabela 6 mostra o cálculo de pesos obtidos para a Figura 31. O desempate é realizado pela soma dos pesos desse *frame* com os restantes dos *frames* encontrados no intervalo configurado.

Tabela 5 – Porcentagens de acertos referentes a cada formação

Formação	Acurácia (%)
5 - 3 - 2	50.0
4 - 5 - 1	25.0
4 - 4 - 2	50.0
4 - 3 - 3	90.0
4 - 2 - 4	60.0
3 - 5 - 2	50.0
3 - 4 - 3	46.0

Fonte: Acervo do autor.

Figura 30 – Exemplo - Erro de classificação causado por classificação de times incorreta



Fonte: Acervo do autor.

Figura 31 – Frame classificado de forma correta



Fonte: Acervo do autor.

Tabela 6 – Cálculo de pesos para a formação 3-3-3.

Formação	Peso
5 - 3 - 2	3
5 - 4 - 1	5
4 - 5 - 1	5
4 - 4 - 2	3
4 - 3 - 3	1
4 - 2 - 4	3
3 - 5 - 2	3
3 - 4 - 3	1

Fonte: Acervo do autor.

5 Conclusão

Este trabalho apresentou a proposta de metodologia que tem como objetivo o reconhecimento de formações táticas em um jogo de futebol, com o uso de processamento de imagem, aprendizado de máquina e fundamentos do esporte.

A etapa de extração de campo, apresentou resultados positivos, porém ela apresenta deficiência se a cor ou a iluminação do gramado apresentarem mudanças, principalmente devido a mudanças climáticas, como o acúmulo de neve no gramado. A fase de reconhecimento de jogadores obteve bons resultados, com um número de falsos positivos aceitáveis.

A fase de classificação foi o período que mais afetou a aplicação da metodologia. A qualidade baixa dos vídeos obtidos influenciou significativamente os resultados nessa etapa. Qualidade apresentada devido a limitação de hardware apresentada na aquisição das amostras. Uma melhora nesse sistema poderia representar um aumento considerável na acurácia dos resultados.

5.1 Trabalhos Futuros

Com um resultado positivo, a metodologia abre espaço para a evolução dos trabalhos apresentados com o desenvolvimento de métodos que possam ser capazes de identificar o momento de jogo em que as formações foram detectadas. Dessa forma, uma equipe poderia perceber falhas ou mudanças em seu posicionamento que tenham afetado o resultado final do jogo. A análise tática de times adversários também pode ser feita com o uso dessa ferramenta. Em campeonatos continentais, equipes nem sempre tem conhecimento sobre o adversário, porém o estudo desse time pode ser realizado com o uso dessa ferramenta.

A obtenção das porcentagens de tempo que cada formação foi utilizada durante a partida também é uma melhora a ser acrescentada a metodologia de forma a permitir mais uma fonte de dados para a análise do jogo.

Os resultados apresentados também demonstram que o uso de vídeos com melhor resolução podem gerar impactos significativos na qualidade da aplicação, pois facilita a separação de times fornecendo um número maior de informações.

A movimentação de câmera apresentada também gerou impactos na resolução do problema, devido a movimentação lateral que a mesma apresentou, que por diversas vezes removeu jogadores do campo de visão. A utilização de câmeras fixas e de qualidade superior seriam uma evolução a ser implementada na metodologia.

Referências

- ALBUQUERQUE, M. P. de Albuquerque; Marcelo Portes de. *Processamento de Imagens: Métodos e Análises*. 2000. <<http://www.cbpf.br/cat/pdsi/pdf/ProcessamentoImagens.PDF>>. Acesso em 23 de Março de 2016. Citado na página 16.
- BAEK, J.; E.JACOBS, D. *Accelerating Spatially Varying Gaussian Filters*. 2011. <<http://pt.slideshare.net/chiaminhsu/study-accelerating-spatially-varying-gaussian-filters>>. Acesso em 17 de Fevereiro de 2016. Citado na página 19.
- CAMARÃO, B. *A utilização do GPS na prática de treinamentos e jogos de futebol*. 2009. <<http://universidadedofutebol.com.br/a-utilizacao-do-gps-na-pratica-de-treinamentos-e-jogos-de-futebol/>>. Acesso em 22 de Março de 2016. Citado na página 13.
- CARVALHO, F. M.; SCAGLIA, A. J.; COSTA, I. T. da. *Influência do Desempenho Tático Sobre o Resultado Final em Jogo Reduzido de Futebol*. 2013. Acesso em 17 de Fevereiro de 2016. Citado 2 vezes nas páginas 13 e 16.
- CHOI, S.; SEO, Y.; KIM, H.; HONG, K.-S. *Where are the ball and players? : Soccer Game Analysis with Color-based Tracking and Image Mosaick*. 1997. <<http://dl.acm.org/citation.cfm?id=686879>>. Acesso em 12 de Dezembro de 2015. Citado 3 vezes nas páginas 14, 27 e 28.
- CONCI, A. *Canny: Detecção de Borda*. 2007. <<http://www2.ic.uff.br/~aconci/canny.pdf>>. Acesso em 12 de Março de 2016. Citado na página 22.
- DAWSON-HOWE, K. *A Practical Introduction to Computer Vision with OpenCV*. 2014. Citado 4 vezes nas páginas 20, 21, 22 e 24.
- DOMINGOS, P. *A Few Useful Things to Know about Machine Learning*. 2012. Citado na página 24.
- FIFA. *265 million playing football*. 2007. <http://www.fifa.com/mm/document/fifafacts/bcoffsurv/emaga_9384_10704.pdf>. Acesso em 16 de Fevereiro de 2016. Citado na página 12.
- FIFA. *2014 FIFA World Cup Television Audience Report*. 2014. <[http://resources.fifa.com/mm/document/affederation/tv/02/74/55/57/2014fwcbraziltvaudiencereport\(draft5\)\(issuedate14.12.15\)_neutral.pdf](http://resources.fifa.com/mm/document/affederation/tv/02/74/55/57/2014fwcbraziltvaudiencereport(draft5)(issuedate14.12.15)_neutral.pdf)>. Acesso em 16 de Fevereiro de 2016. Citado na página 12.
- FISHER, R.; PERKINS, S.; WALKER, A.; WOLFART., E. *Gaussian Smoothing*. 2003. <<http://homepages.inf.ed.ac.uk/rbf/HIPR2/gsmooth.htm>>. Acesso em 08 de Março de 2016. Citado na página 19.
- GRECO, P.; MESQUITA, I.; MÜLLER, E.; SILVA, B.; GARGANTA, D. C. J.; COSTA, I. *Análise do Comportamento Tático de Jogadores de Futebol através da aplicação do Teste “Gr3-3Gr”, em dois períodos de jogo distintos*. 2009. Acesso em 16 de Fevereiro de 2016. Citado na página 13.

- HARRINGTON, P. *Machine Learning in Action*. 2012. Citado na página 24.
- HUANG, Y.; LLACH, J.; BHAGAVATHY, S. *Players and Ball Detection in Soccer Videos Based on Color Segmentation and Shape Analysis*. 2007. Citado na página 14.
- JAMUNDA, T. *Reconhecimento de Formas: A Transformada de Hough*. 2000. Acesso em 10 de Março de 2016. Citado na página 23.
- KHATOONABADI, S. H.; RAHMATI, M. *Automatic soccer players tracking in goal scenes by camera motion elimination*. 2009. <<http://www.emarketer.com/Article/2-Billion-Consumers-Worldwide-Smartphones-by-2016/1011694>>. Acesso em 12 de Dezembro de 2015. Citado 3 vezes nas páginas 14, 27 e 29.
- LAGANIÈRE, R. *OpenCV 2 Computer Vision Application Programming Cookbook*. 2011. Citado na página 18.
- MACEDO, M. M. G. de. *Uso da Transformada de Hough na Vetorização de Moldes e Outras Aplicações*. 2005. Acesso em 14 de Março de 2016. Citado na página 23.
- MATHWORKS. *Classification Using Nearest Neighbors*. 2016. <<http://www.mathworks.com/help/stats/classification-using-nearest-neighbors.html>>. Citado na página 25.
- MAĆKOWIAK, S.; KONIECZNY, J.; KURC, M.; MAĆKOWIAK, P. *Football player detection in video broadcast*. 2010. Citado na página 14.
- MENEGUETTE, A. *Cartografia Temática*. 2009. Citado na página 18.
- MOESLUND; B., T.; GRAHAM, T.; ADRIAN, H. *Computer Vision in Sports*. 2014. Citado na página 14.
- MORSE, B. S.; UNIVERSITY, B. Y. *Thresholding*. 2000. Acesso em 12 de Março de 2016. Citado na página 18.
- OPENCV.ORG. *Image Filtering*. 2014. <<http://docs.opencv.org/3.0-beta/modules/imgproc/doc/filtering.html>>. Acesso em 03 de Março de 2016. Citado na página 20.
- OPENCV.ORG. *Smoothing Images*. 2014. <http://docs.opencv.org/2.4/doc/tutorials/imgproc/gaussian_median_blur_bilateral_filter/gaussian_median_blur_bilateral_filter.html>. Acesso em 16 de Fevereiro de 2016. Citado na página 18.
- PARREIRA, C. A. *Evolução Tática E Estratégias De Jogo*. São Paulo, Brasil: Escola Brasileira de Futebol, 2005. Citado na página 12.
- RIBEIRO, F. *Esquemas táticos*. 2013. <<http://crescieagora.com.br/futebol/esquema-tatico/>>. Acesso em 17 de Fevereiro de 2016. Citado na página 17.
- ROCHA, R. A. *Análise Da Evolução Dos Esquemas Táticos Do Futebol Brasileiro*. 2010. Acesso em 16 de Fevereiro de 2016. Citado na página 12.
- SALEM, O. G. *Connected Component Labeling Algorithm*. 2014. Citado na página 22.
- SAMET, H.; TAMMINEN, M. *Efficient Component Labeling of Images of Arbitrary Dimension Represented by Linear Bintree*s. 1988. Citado na página 22.

- SILVA, D. S. da. *Aplicação de Transformadas de Hough na Detecção de Leveduras Viáveis e Inviáveis*. 2010. Acesso em 12 de Março de 2016. Citado na página 24.
- SONG, Y.; HUANG, J.; ZHOU, H. Z. D.; GILES, C. L. *IKNN: Informative K-Nearest Neighbor Pattern Classification*. 2007. Acesso em 18 de Março de 2016. Citado na página 25.
- SUZUKI, S.; ABE, K. *Topological Structural Analysis of Digitized Binary Images by Border Following*. 1985. Citado na página 29.
- SZELISKI, R. *Computer Vision: Algorithms and Applications*. 2011. Citado na página 18.
- THIRUMURUGANATHAN, S. *A Detailed Introduction to K-Nearest Neighbor (KNN) Algorithm*. 2010. Acesso em 18 de Março de 2016. Citado na página 25.
- VENDITE, C. C.; MORAES, A. C. de. *Sistema, Estratégia e Tática de Jogo no Futebol: Análise do Conhecimento dos Profissionais que Atuam no Futebol*. 2006. Acesso em 16 de Fevereiro de 2016. Citado na página 16.
- WILSON, J. *Inverting the Pyramid*. 2013. Acesso em 17 de Fevereiro de 2016. Citado na página 16.