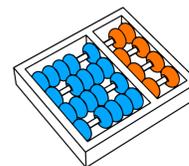


Erikson Freitas de Moraes

**“Rastreamento de Jogadores de Futebol de Salão  
Usando Filtro de Partículas e Coordenadas no Plano  
da Quadra.”**

CAMPINAS  
2012





Universidade Estadual de Campinas  
Instituto de Computação

**Erikson Freitas de Moraes**

**“Rastreamento de Jogadores de Futebol de Salão  
Usando Filtro de Partículas e Coordenadas no Plano  
da Quadra.”**

Orientador(a): **Prof. Dr. Siome Klein Goldenstein**

Co-Orientador(a): **Prof. Dr. Anderson de Rezende Rocha**

Tese de Doutorado apresentada ao Programa de Pós-Graduação em Ciência da Computação do Instituto de Computação da Universidade Estadual de Campinas para obtenção do título de Doutor em Ciência da Computação.

ESTE EXEMPLAR CORRESPONDE À VERSÃO  
FINAL DA TESE DEFENDIDA POR ERIKSON  
FREITAS DE MORAIS, SOB ORIENTAÇÃO  
DE PROF. DR. SIOME KLEIN GOLDENS-  
TEIN.

---

Assinatura do Orientador(a)

CAMPINAS

2012

iii

FICHA CATALOGRÁFICA ELABORADA POR  
ANA REGINA MACHADO - CRB8/5467  
BIBLIOTECA DO INSTITUTO DE MATEMÁTICA, ESTATÍSTICA E  
COMPUTAÇÃO CIENTÍFICA - UNICAMP

M792r Morais, Erikson Freitas de, 1978-  
Rastreamento de jogadores de futebol de salão usando filtro de partículas e coordenadas no plano da quadra / Erikson Freitas de Morais. – Campinas, SP : [s.n.], 2012.

Orientador: Siome Klein Goldenstein.  
Coorientador: Anderson de Rezende Rocha.  
Tese (doutorado) – Universidade Estadual de Campinas,  
Instituto de Computação.

1. Rastreamento automático. 2. Visão por computador. 3. Futebol de salão. I. Goldenstein, Siome Klein, 1972-. II. Rocha, Anderson de Rezende. III. Universidade Estadual de Campinas. Instituto de Computação. IV. Título.

Informações para Biblioteca Digital

**Título em inglês:** Tracking futsal players using particle filter and court plane coordinates

**Palavras-chave em inglês:**

Automatic tracking

Computer vision

Indoor soccer

**Área de concentração:** Ciência da Computação

**Titulação:** Doutor em Ciência da Computação

**Banca examinadora:**

Siome Klein Goldenstein [Orientador]

Milton Shoiti Misuta

João Paulo Papa

Felipe Arruda Moura

Jaques Wainer

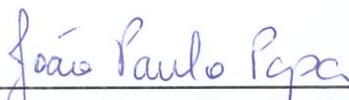
**Data de defesa:** 12-12-2012

**Programa de Pós-Graduação:** Ciência da Computação

## TERMO DE APROVAÇÃO

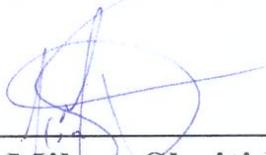
Tese Defendida e Aprovada em 12 de Dezembro de 2012, pela

Banca examinadora composta pelos Professores Doutores:



---

**Prof. Dr. João Paulo Papa**  
FC / UNESP



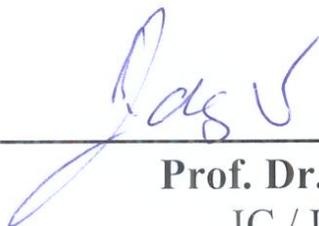
---

**Prof. Dr. Milton Shoiti Misuta**  
FCA / UNICAMP



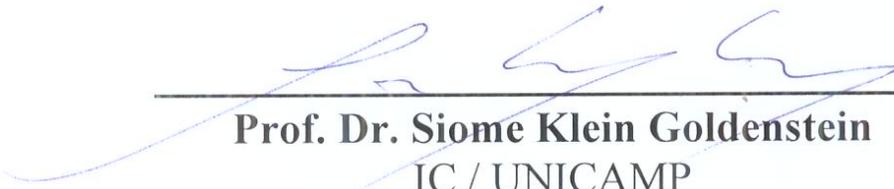
---

**Prof. Dr. Felipe Arruda Moura**  
CEFE / UEL



---

**Prof. Dr. Jacques Wainer**  
IC / UNICAMP



---

**Prof. Dr. Siome Klein Goldenstein**  
IC / UNICAMP



# Rastreamento de Jogadores de Futebol de Salão Usando Filtro de Partículas e Coordenadas no Plano da Quadra.

Erikson Freitas de Morais<sup>1</sup>

12 de Dezembro de 2012

## Banca Examinadora:

- Prof. Dr. Siome Klein Goldenstein (Supervisor/*Orientador*)
- Prof. Dr. Milton Shoiti Misuta  
FCA - UNICAMP
- Prof. Dr. Jacques Wainer  
IC - UNICAMP
- Prof. Dr. João Paulo Papa  
FC - UNESP
- Prof. Dr. Felipe Arruda  
CEFE - UEL

---

<sup>1</sup>Suporte financeiro de: Bolsa do CNPq (processo 141054/2010-7) 2010–2012 e CAPES (processo 01-P-04388-2010) 2008–2010.



# Abstract

Nowadays, it is common to have news of athletes that go beyond their physical limits. Meanwhile, their technical teams watch and analyze the causes and results of their performance in order to find somewhere that can be improved. The observation and analysis of the events involved in a game of futsal can identify failure and with this, design improvements. In the sport case, the improvements appear as overcoming limits, tactics more efficient and identify the shortcomings of the opponent. The use of video cameras to record sports team games allows the detailed performance evaluation of physical and tactical team on the pitch. the videos analyzed in the laboratory reveal failure very difficult to detect during the game as much tactical interest as physical preparation point of view. Estimating the positions of players at each instant of the video allows calculate an approximation of players trajectories. This is the start point to recover a set of important information such as distance total traveled, average speed and instantaneous speed, moments acceleration, among others. In this sense, techniques of Computer Vision has been used to analyze tactics and physiological games team sports. Using the application of visual tracking, at the end of video is possible to check important data such as average and maximum speeds of players during the game, total and partial distances covered, and especially the trajectories traveled by the players on the pitch. In this work, we discussed the problem of visual tracking of multiple objects of interest, focusing on futsal players during the game. This problem has attracted the interest of the scientific community both tactical and physiological point of view as the computational point of view. The problem of tracking players in a futsal game handles objects of interest with dynamic motion very active, which makes the problem complex. However, solutions for tracking people in a environment as dynamic as a futsal game has applications in different areas such as security environments, monitoring lab animals and traffic pedestrian. We treat our problem in a scenario composed of four digital video cameras positioned strategically around the pitch. We used machine learning techniques to detect players in pictures and then merged detections on a virtual plane that corresponds to the plane of the pitch. Using an appearance model coupled to a particle filter we calculated good approximations of players positions and consequently we track using real coordinates. In our work,

we found good results with little human intervention and training relatively simple.

# Resumo

Nos dias de hoje, é comum termos notícia de atletas que ultrapassam seus limites físicos. Enquanto isso, suas equipes técnicas observam e analisam as causas e os resultados de seu desempenho com o objetivo de encontrar algum ponto que possa ser melhorado. A observação e análise dos eventos envolvidos em um jogo de futebol de salão permite identificar falhas e com isso projetar melhorias. No caso do esporte, as melhorias aparecem na forma de superação de limites, táticas mais eficientes e identificação das deficiências do adversário. A utilização de câmeras de vídeo para gravar partidas de esportes coletivos, permite a avaliação minuciosa do comportamento físico e tático do time dentro de campo. Os vídeos analisados em laboratório revelam falhas muito difíceis de serem detectadas durante o jogo, tanto de cunho tático quanto do ponto de vista da preparação física. Estimar as posições dos jogadores em cada instante do vídeo permite calcular uma aproximação das trajetórias dos jogadores. Esse é o ponto de partida para recuperar um conjunto importante de informações como distância total percorrida, velocidade média e instantâneas, velocidade máxima, momentos de aceleração, entre outras. Neste sentido, técnicas de *Visão Computacional* tem sido utilizadas na análise tática e fisiológica em partidas de esportes coletivos. Por meio da aplicação de rastreamento visual, ao final do vídeo é possível verificar dados importantes como velocidades média e máxima dos jogadores durante a partida, distâncias percorridas total e parciais, e principalmente as trajetórias percorridas pelos jogadores dentro de campo. Neste trabalho abordamos o problema de rastreamento visual de múltiplos objetos de interesse, com foco no rastreamento de jogadores de futsal durante uma partida. Este problema tem despertado o interesse da comunidade científica tanto do ponto de vista tático e fisiológico, quanto do ponto de vista computacional. O problema de rastrear jogadores em uma partida de futsal lida com objetos de interesse com dinâmica de movimento bastante ativa, o que torna o problema complexo. No entanto, soluções para o rastreamento de pessoas em um ambiente tão dinâmico quanto um jogo de futsal tem aplicações em diferentes áreas como segurança de ambientes, monitoramento de cobaias, tráfego de pedestres, etc. Tratamos nosso problema em um cenário composto por quatro câmeras de vídeo digital posicionadas estrategicamente ao redor do quadra. Usamos técnicas de aprendizado de máquina para

detectar os jogadores nas imagens resultantes e em seguida fundimos as detecções em um plano virtual que corresponde ao plano da quadra. Usando um modelo de aparência acoplado a um filtro de partículas calculamos boas aproximações das posições dos jogadores e conseqüentemente rastreamos usando coordenadas reais. Em nosso trabalho encontramos bons resultados com pouca intervenção humana e um treinamento relativamente simples.

# Agradecimentos

Foi muito difícil chegar ao final do doutorado. Mas não foi impossível graças a uma força maior que não posso explicar de onde vem nem como atua, mas tenho minha história como testemunho.

Vim de uma cidade pequena e provinciana. A aprovação para o doutorado foi providencial para o momento em que vivíamos. Então, vim para Campinas com um objetivo bem estabelecido e deixei minha família para trás. Vicente Amorim, amigo e ex-aluno, me recebeu em sua casa onde pude fazer bons amigos como César, André, Pedro, Guilherme, Thomas, Leonardo, Rodrigo e Arnaldo que entendiam minhas necessidades, respeitaram minhas dificuldades e me ensinaram novas soluções.

Na UNICAMP, tive o privilégio de me tornar aluno de um professor que, um dia em sua sala, me disse: “Você aqui é minha responsabilidade” me ajudando a resolver problemas pessoais. Lídia tinha razão quando me disse que as pessoas certas aparecem em minha vida, no momento certo. Siome foi o professor certo no momento certo, de decisões e indicações acertadas. Assim como Anderson, que era um colega de laboratório e “irmão” de orientação, se tornou meu co-orientador dedicado e exigente. A participação de vocês no meu processo de doutoramento foi da mais alta importância. Aprendi muito com os dois e vou continuar aprendendo, seja em minhas lembranças, seja em nossas parcerias daqui por diante. É bom saber que tenho onde me espelhar.

Sou grato também ao professor Ricardo Barros que nos recebeu no LIB-FEF, de onde vieram as oportunidades de pesquisa, as ideias, as discussões com os orientadores, os dados para o trabalho e como consequência de tudo isso, minha tese. Lá, também pude conhecer pessoas importantes que exerceram influência direta em minha pesquisa como Milton Misuta e Fernando Moura, que tiveram muita paciência em me mostrar como era seu ambiente de trabalho e, assim, permitiram o surgimento de minhas próprias ideias.

Cada disciplina foi um desafio a parte. Cada desafio, vivido ao lado de figuras marcantes que vou sempre lembrar, cada um por suas características próprias. Não vou esquecer de acordar cedo para as aulas de Arquitetura e seguir o caminho junto com Carlos Armindo, estudar durante os finais de semana junto com Fábio Faria, Maria Angélica, Juliana Galvani, Luciano Chaves, Juliana de Santi, Leonardo Miranda, Gabriel Cavalcante

e tantos outros. Também não posso esquecer de Adilson Bonifácio que dividiu comigo a sala 92 na grande maioria do tempo que estive por lá.

Durante o doutorado, eu e minha esposa, passamos por uma gravidez de risco, que exigiu cuidados médicos e repouso absoluto para ela durante aproximadamente 8 dos 9 meses. Estivemos sem nossas famílias, primeiro pela distância e depois porque estavam ocupados cuidando de seus respectivos pacientes: meu sogro com um infarto seguido da síndrome de Guillain-Barré e meu pai com um infarto. Aqueles que conviveram comigo sabem como foram difícil aqueles meses. Foi quando ouvi de uma pessoa muito importante: “Vai cuidar de sua família, o resto vemos depois.”

Tive o amparo de meus amigos de república que se revezaram por iniciativa própria para nos ajudar. Qualquer agradecimento que eu fizer a Leonel, Marlon e Jefersson e também a Juliana de Sant e Adilson Bonifácio, será insignificante perto do que fizeram por nós naqueles meses. Já o amparo médico veio da própria universidade. Foi no CAISM-UNICAMP que o problema de minha esposa foi detectado, tratado, acompanhado e onde minha filha Alice nasceu com toda saúde, uma raridade para o caso dela. Hoje sabemos que se fosse em nossa cidade o final teria sido diferente.

Minha família foi muito paciente e compreensiva comigo durante todo esse tempo. Me deram força, entenderam minha ausência e permitiram meus momentos de isolamento. Reconheço as dificuldades em tudo isso.

A todos, o meu muito obrigado. Obrigado por terem participado deste processo difícil no momento certo e a seu modo. Consegui chegar ao fim porque Deus sempre esteve comigo, retirando de meu caminho as pedras maiores e colocado vocês junto de mim.

# Sumário

<b>Abstract</b>	<b>ix</b>
<b>Resumo</b>	<b>xi</b>
<b>Agradecimentos</b>	<b>xiii</b>
<b>1 Introdução</b>	<b>1</b>
1.1 Motivação . . . . .	3
1.2 Abordagem . . . . .	5
1.3 Organização do texto . . . . .	8
<b>2 Trabalhos Relacionados</b>	<b>11</b>
2.1 Detecção de Jogadores . . . . .	11
2.2 Rastreamento de Jogadores . . . . .	14
2.3 Modelo de aparência . . . . .	16
<b>3 Localização no Plano da Quadra</b>	<b>21</b>
3.1 Descrição da Técnica . . . . .	22
3.1.1 Detecção Independente no Plano da Imagem . . . . .	22
3.1.2 Projeção da Localização em Coordenadas de Imagem no Plano da Quadra . . . . .	25
3.1.3 Fusão dos Dados Multi-Câmera . . . . .	28
3.2 Metodologia e Experimentos . . . . .	31
<b>4 Rastreamento de Jogadores no Plano da Quadra</b>	<b>39</b>
4.1 Conceitos Básicos . . . . .	40
4.2 Método proposto . . . . .	43
4.2.1 Reamostragem . . . . .	43
4.2.2 Propagação . . . . .	45
4.2.3 Função de Observação . . . . .	46

4.3	Metodologia e experimentos . . . . .	48
<b>5</b>	<b>Modelo de Aparência</b>	<b>57</b>
5.1	Aparência do Objeto . . . . .	58
5.2	O Modelo na Câmera . . . . .	60
5.3	Função Multimodal Fortalecida . . . . .	61
5.4	Metodologia e Experimentos . . . . .	63
5.4.1	Experimentos Usando do Modelo de Aparência . . . . .	66
<b>6</b>	<b>Conclusões</b>	<b>71</b>
6.1	Trabalhos Futuros . . . . .	73
<b>A</b>	<b>Gráficos dos erros médios acumulados para a localização dos jogadores no plano da quadra</b>	<b>83</b>

# Lista de Tabelas

3.1	Contagem de DC, FP e FN . . . . .	35
3.2	Erros médios descritos em metros . . . . .	36
3.2	Erros médios descritos em metros . . . . .	37
4.1	Erros médios e desvios encontrados para as trajetórias completas encontradas pelo rastreador. . . . .	55
5.1	Erros médios( $\mu$ ) e desvios( $\sigma$ ) padrão encontrados para trajetórias completas encontradas pelo rastreador <b>sem o modelo de aparência</b> . . . . .	68
5.2	Erros e desvios padrão encontrados para trajetórias completas encontradas pelo rastreador <b>com o modelo de aparência</b> . . . . .	69



# Lista de Figuras

1.1	Posicionamento das câmeras ao redor do quadra de um jogo de futebol de salão . . . . .	4
1.2	Exemplo de quadros de vídeo capturados . . . . .	5
1.3	Mapa de pontos característicos . . . . .	6
1.4	Etapas principais do trabalho . . . . .	7
3.1	Diagrama com os estágios principais do modelo proposto para localização de jogadores . . . . .	23
3.2	Exemplos de características do tipo dois, três e quatro retângulos . . . . .	24
3.3	Exemplo de Imagem Integral <i>ii</i> . . . . .	25
3.4	Cascata de classificadores . . . . .	25
3.5	Função de probabilidade 2D . . . . .	29
3.6	Curvas de nível ao redor dos pontos projetados . . . . .	32
3.7	Curvas de nível obtidas por um fator de escala . . . . .	33
3.8	Erro médio na estimativa . . . . .	34
4.1	Passos do filtro de partículas. . . . .	42
4.2	Exemplo de reamostragem. . . . .	44
4.3	Estágios principais do modelo de observação. . . . .	46
4.4	Observação do sistema de rastreamento em um dos ciclos do filtro de partículas. . . . .	48
4.5	Inicialização dos Rastreadores . . . . .	49
4.6	Demonstração do rastreamento . . . . .	50
4.6	Exemplo de confusão por proximidade de jogadores. . . . .	52
4.7	Trajetórias com erro médio, maior e menor encontradas pelo rastreador. . . . .	53
4.8	Erros médios encontrados em trajetórias bem sucedidas. . . . .	54
5.1	Modelo de aparência para um jogador observado . . . . .	59
5.2	Gaussianas com pesos diferentes . . . . .	62
5.3	Demonstração do rastreamento com o modelo de aparência . . . . .	64
5.3	Trajetória encontrada pelo rastreador sem o modelo de aparência . . . . .	66

5.4	Trajétória encontrada pelo rastreador usando o modelo de aparência . . . .	67
5.5	Erros médios acumulados nas trajetórias bem sucedidas sem o modelo de aparência . . . . .	68
5.6	Erros médios acumulados nas trajetórias bem sucedidas com o modelo de aparência . . . . .	69
A.1	Erro médio na estimativa . . . . .	83

# Capítulo 1

## Introdução

Nos dias atuais, a busca por superação é uma constante para a humanidade. O significado dessa afirmação fica evidente quando observamos atletas olímpicos. A cada quatro anos, novos recordes são estabelecidos. Os atletas ultrapassam seus limites físicos enquanto suas equipes observam e analisam as causas e os resultados de seu desempenho com o objetivo de encontrar algum ponto que possa ser melhorado. Qualquer detalhe modificado, por menor que seja, pode representar a vitória no final da competição. É o caso, por exemplo, dos atletas de natação utilizando roupas confeccionadas com materiais e métodos capazes de diminuir o atrito com a água. Exemplos parecidos podem ser encontrados em várias modalidades de esportes, sempre baseadas em observações realizadas pela equipe técnica durante várias competições.

A observação e análise dos eventos envolvidos em um jogo permite identificar falhas e, com isso, projetar melhorias. A observação minuciosa feita pelo olhar de um especialista leva a um estudo sistemático de detalhes imperceptíveis para pessoas não treinadas. Observando e analisando, novas propostas e métodos são elaborados com o objetivo de trazer melhorias para o processo analisado [48, 52, 53, 55, 57, 61, 63]. No caso do esporte, as melhorias aparecem na forma de superação de limites, táticas mais eficientes e estudo das deficiências do adversário.

A evolução tecnológica presente nas diversas modalidades do esporte desperta, cada vez mais, a preocupação com melhoria dos resultados [12, 18, 24, 28, 33]. Nesse sentido, as equipes técnicas buscam meios e ferramentas que lhes permitam a observação e análise detalhada da performance de seus atletas. Com a popularização das câmeras de vídeo, a gravação das partidas de esportes coletivos se tornou uma ferramenta de grande importância técnica para os treinadores. Uma vez gravadas as partidas, as equipes podem analisar o desempenho de seus atletas, avaliar seus métodos de treinamento e observar o comportamento de seus adversários [6, 11, 18].

A utilização de câmeras de vídeo para gravar partidas de esportes coletivos permite a

avaliação minuciosa do comportamento físico e tático do time dentro de campo. Os vídeos analisados em laboratório revelam falhas muito difíceis de serem detectadas durante o jogo, tanto de cunho tático quanto do ponto de vista da preparação física [6, 12, 16, 28, 38].

As partidas gravadas também servem para avaliação dos oponentes. A observação do comportamento característico do oponente permite à equipe técnica reavaliar seu treinamento tático antes da competição. Essas conclusões são alcançadas com base na observação crítica de jogos, analisando medidas como estimativas de velocidades, áreas de abrangência dos jogadores dentro de campo, número de erros e acertos dos jogadores, especialidades de cada um, entre outros.

Nos vídeos estão gravados os erros e acertos das equipes que se confrontaram. Uma equipe especializada pode avaliar os vídeos e anotar os erros cometidos para posteriormente classificá-los de acordo com seu interesse. Por exemplo, depois de anotados os eventos de um jogo, é possível visualizar gráficos que mostram a área de cobertura de cada jogador, o número de passes errados por equipe ou por jogador, jogadas bem sucedidas e as tentativas frustradas identificando, nesses casos, o que levou ao insucesso. De posse das anotações dos vídeos de um jogo, a equipe técnica possui um universo de informações valiosas com grande potencial de ajudar em seu trabalho.

Além de conter informações úteis para a avaliação tática das equipes, a análise dos vídeos das partidas pode revelar informações úteis também do ponto de vista do treinamento físico [41, 42, 44, 48–50]. Esse tipo de informação permite o dimensionamento do treinamento físico buscando otimizar os ganhos sem extrapolar os limites necessários ao bom condicionamento dos atletas individualmente e conseqüentemente do time. Estimar as posições dos jogadores em cada instante do vídeo constitui um caminho para calcular uma aproximação das trajetórias dos jogadores. Esse é o ponto de partida para recuperar um conjunto importante de informações como distância total percorrida, velocidade média e instantâneas, velocidade máxima, momentos de aceleração, entre outras.

Com base nas trajetórias obtidas ao final dos jogos, outras informações são calculadas. Sabendo as posições de um jogador ao longo do tempo, podemos encontrar a distância total percorrida e traçar gráficos de velocidades e acelerações. Tais informações são importantes para auxiliar a equipe técnica do time em suas tomadas de decisão durante o treinamento.

O registro das partidas em vídeo permite, ainda, a gravação em múltiplos pontos de vista utilizando várias câmeras simultaneamente. Com isso, alguma informação que não possa ser observada por uma das câmeras, provavelmente será registrada em outra. O registro redundante das informações visuais possibilita extrair informações com maior confiança e menores níveis de erros.

Observar diretamente os vídeos traz ganhos significativos, mas é possível ganhar ainda mais com a ajuda de métodos computacionais. Neste sentido, técnicas de *Visão Computacional* tem sido utilizadas na análise tática e de desempenho físico em partidas de

futebol [17, 18, 32, 35], hoquei [34], handball [7], basquetebol [2]. Por meio da aplicação de rastreamento visual, ao final do vídeo é possível verificar dados importantes como velocidades média e máxima dos jogadores durante a partida, distâncias percorridas total e parciais e, principalmente, as trajetórias percorridas pelos jogadores dentro de campo.

O rastreamento visual de objetos é definido por Gevarter [22] como: *processar uma sequência de imagens para descrever o movimento de um ou mais objetos em uma cena*. Outra definição dada por Forsyth e Ponce [19] consiste em: *gerar inferência sobre o movimento de um objeto dada uma sequência de imagens*. A partir das duas definições, é possível perceber que o ato de rastrear consiste em *inferir* informações relacionadas no espaço e no tempo, analisando imagens em sequência de uma cena e com interesse em um ou mais objetos, tal como jogadores em uma partida de futebol de salão.

Neste trabalho, abordamos o problema de rastreamento visual de múltiplos objetos de interesse, com foco no rastreamento de jogadores de futebol de salão durante uma determinada partida. Este problema tem despertado o interesse da comunidade científica tanto do ponto de vista tático e fisiológico, quanto do ponto de vista computacional. O problema de rastrear jogadores em uma partida de futebol de salão lida com objetos de interesse com dinâmica de movimento bastante ativa, o que torna o problema complexo. Soluções para o rastreamento de pessoas em um ambiente tão dinâmico quanto um jogo de futebol de salão podem ser aplicadas em áreas diferentes como segurança de ambientes, monitoramento de cobaias, contagem de pedestres, etc.

## 1.1 Motivação

Tão importante quanto o talento dos jogadores, a análise tática em esportes coletivos pode ajudar as equipes a alcançar seus objetivos durante um jogo. Essas análises ajudam o treinador a explorar as qualidades do time, as fraquezas dos adversários, corrigir erros táticos, verificar as condições físicas dos jogadores, dimensionar o treinamento físico, entre outros. Para isso, a informação de posição dos jogadores durante o jogo é necessária. Um sistema de Visão Computacional que realiza o rastreamento dos jogadores pode ajudar nessa questão.

Treinadores de futebol de salão e instituições educacionais esportivas estão interessados nessa análise baseada em dados obtidos por rastreamento. Isso ajuda os times a explorar de forma mais eficiente o espaço do jogo.

O futebol de salão tem grande importância no esporte mundial dos dias de hoje. Por isso, tem sido alvo de interesse técnico e científico apresentando diversas aplicações dedicadas à análises tática, física e de fatores fisiológicos. Em todos os casos, a análise tem como base as trajetórias obtidas pelo o rastreamento dos jogadores durante a partida. As equipes táticas estão interessadas nas trajetórias dos jogadores para verificar a eficiência de

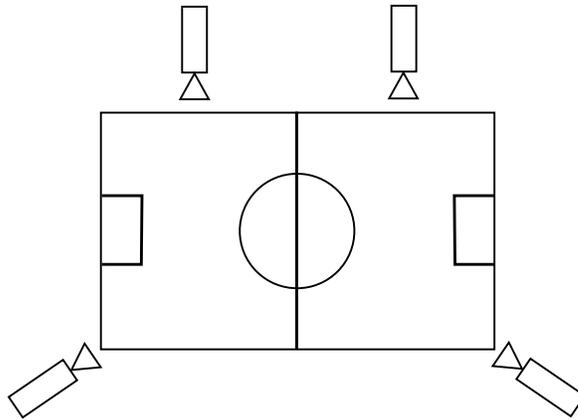


Figura 1.1: Posicionamento das câmeras ao redor do quadra de um jogo de futebol de salão. O campo de visão de cada câmera é configurado de forma que seja coberto por inteiro um dos lados da quadra, permitindo sobreposições. Nessa configuração, cada jogador está sempre no campo de visão de pelo menos duas câmeras.

seus posicionamentos dentro do jogo. As equipes técnicas, responsáveis pelo treinamento físico dos atletas, analisam as trajetórias para verificar dados como velocidades alcançadas, picos de aceleração e distância percorrida bem como para dimensionar o treinamento físico dos atletas. A análise fisiológica utiliza dados obtidos com base nas trajetórias para avaliar níveis de esforço e auxiliar no dimensionamento técnico dos atletas.

A aplicação de métodos de Visão Computacional na análise de partidas de esportes coletivos tem se mostrado útil, também, do ponto de vista científico. O ambiente dinâmico de um jogo coletivo permite experimentar e validar métodos interessantes não apenas para as aplicações pertinentes ao esporte, mas que se adequem a outros casos como segurança de interiores, contagem de pedestres, monitoramento e muitos outros.

Outros esportes coletivos como futebol [18, 32, 35], hóquei [34], handball [7] e basquetebol [2] foram alvo de pesquisa usando métodos de Visão Computacional para análise tática ou fisiológica. Em nosso trabalho, estamos interessados no rastreamento dos jogadores de futebol de salão, por meio de técnicas de visão computacional e aprendizado de máquina, com o objetivo final de minimizar a intervenção humana no processo de rastreamento. Com isso é possível encontrar trajetórias mais precisas, um resultado importante já que é base para outras informações de grande relevância para as equipes de treinamento técnico e tático.



Figura 1.2: Exemplo de quadros de vídeo capturados. Podemos observar um exemplo de imagem capturada por cada uma das câmeras instaladas. Com base nas imagens, é possível observar as áreas de interseção entre seus campos de observação.

## 1.2 Abordagem

Para tratar nosso problema, utilizamos um cenário constituído por quatro câmeras de vídeo digital similares, posicionadas ao redor de uma quadra de futebol de salão, tal como mostra a Figura 1.1. As quatro câmeras são posicionadas de tal forma que exista alguma sobreposição entre seus campos de visualização, de forma a garantir que qualquer ponto da quadra seja monitorado por pelo menos duas das câmeras. Na Figura 1.2 podemos observar um exemplo de imagem capturada por cada uma das câmeras instaladas ao redor da quadra. Além disso, as coordenadas de pontos estratégicos no plano da quadra são conhecidas e gravadas em um mapa. Com esse mapa, é possível construir uma matriz de transformação baseada em correspondências de pontos da imagem e do plano da quadra para mapear posições no plano da imagem em posições no plano da quadra ou vice-versa. A Figura 1.3 mostra o mapa de pontos característicos da quadra numerados de 1 a 38.

Gravações múltiplas, tomadas de pontos de observação diferentes, podem armazenar redundâncias importantes para métodos de processamento automático e análises futuras. Em nosso caso, estimativas de posição são melhoradas usando observações múltiplas em um processo de fusão de dados, resultando em estimativas mais confiáveis.

Dessa forma, nosso trabalho pode ser dividido em quatro partes principais:

1. aquisição dos dados e sincronização do vídeos;
2. detecção e localização dos objetos de interesse;
3. rastreamento para a extração de trajetórias e
4. modelo de aparência para fortalecer a observação do rastreador.

No primeiro passo, a aquisição é realizada utilizando câmeras digitais FULL-HD e armazenadas em arquivos do tipo TOD. Por conveniência, maior facilidade na manipulação e redução no tempo de processamento dos testes, os vídeos foram convertidos para arquivos do tipo AVI de  $720 \times 480$  *pixels* com a compactação original. A sincronização dos vídeos é feita manualmente em laboratório, utilizando um *software* simples para edição

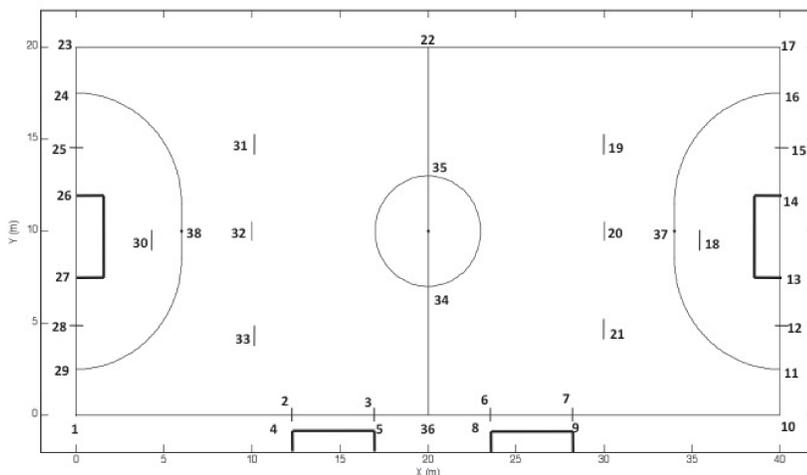


Figura 1.3: Mapa de pontos característicos. O mapa da figura mostra, enumerados, os pontos característicos da quadra cujas posições são conhecidas. O conhecimento desses pontos permite a construção de uma matriz de transformação capaz de mapear pontos no plano da imagem para pontos no plano da quadra.

de vídeo como Avidemux. Esse *software* é de distribuição livre e fácil utilização. A sincronização consiste em gravar em arquivo o número do *frame* que coincide o início do rastreamento em cada uma das câmeras. Esses dados são usados pelo rastreador durante a inicialização do rastreamento.

A detecção utiliza aprendizado de máquina para encontrar os objetos de interesse no plano das imagens. Essas detecções encontradas nas múltiplas câmeras são fundidas em uma função multimodal que representa a probabilidade de um jogador ser encontrado em uma dada posição da quadra. Essa função é uma representação estatística da ocupação da quadra em um dado momento do jogo.

O rastreamento usa essa função multimodal como observação de um filtro de partículas iniciado para um objeto de interesse (no caso um jogador dentro da quadra). A posição inicial do jogador é dada manualmente e a partir disso, o rastreador de cada jogador faz as devidas inicializações.

A função multimodal é usada como função de observação, porém não consegue distinguir os jogadores pois fornece apenas a probabilidade de algum jogador ser encontrado em uma posição da quadra. Por isso, um modelo de aparência precisa ser introduzido com o objetivo de ponderar o resultado da detecção de acordo com o jogador que está sendo rastreado. Com isso, é possível produzir uma função multimodal com picos maiores nas posições que representam detecções mais parecidas com o jogador rastreado.

Podemos resumir as etapas principais do trabalho no diagrama da Figura 1.4. Os vídeos são analisados por detectores treinados para encontrar jogadores. O resultado é

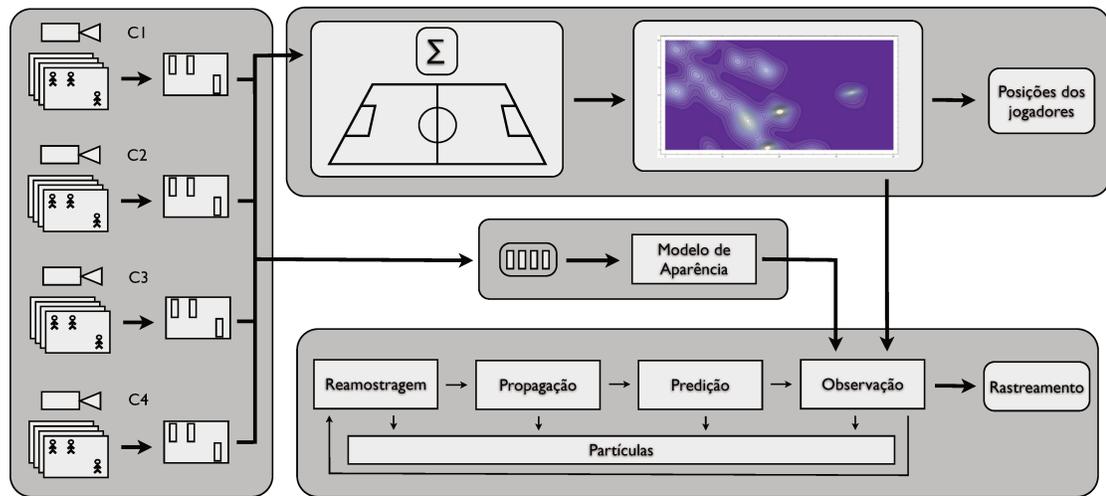


Figura 1.4: Etapas principais do trabalho. O diagrama apresenta as etapas do trabalho e suas ligações. Esse é um processo dinâmico e os estados dos módulos são atualizados usando informações obtidas em outros módulos.

uma função de observação em coordenadas no plano da quadra, usada como observação em um filtro de partículas ponderada por um modelo de aparência. O modelo de aparência é atualizado a cada ciclo do rastreador com base na função multimodal, nas detecções encontradas nas imagens e nas posições encontradas pelo rastreamento. O resultado final do processo são as posições que representarão a trajetória.

O uso do rastreamento visual gera um conjunto de informações que permite uma visualização diferenciada dos jogadores ao longo do tempo, ou a cada instante de tempo usando uma representação que chamamos de estados do objeto. O estado de um objeto rastreado guarda parâmetros que identificam a configuração de um único objeto em um instante de tempo, tais como estimativas da posição e da velocidade instantânea. O fato de serem informações relacionadas ao tempo torna possível rastrear informações que não podem ser detectadas diretamente, como a velocidade e consequentemente aceleração. Para ser encontrada, a informação de velocidade pode ser calculada com base nas informações de estados adjacentes.

Outra informação não capturada diretamente que pode ser interessante para a equipe de treinamento técnico é a área do campo coberta por um atleta durante o jogo. Integrando as posições dos estados de um jogador ao longo do tempo, tem-se a trajetória percorrida e, conseqüentemente, a área do campo coberta por ele. Adicionalmente, pode-se criar um mapa de incidência que mostra as regiões de maior e menor atuação por parte do jogador durante a partida. Essas informações trazem à equipe técnica uma visão privilegiada do jogo e todas são dependentes do rastreamento dos múltiplos jogadores durante o jogo.

### 1.3 Organização do texto

Em nosso texto, dedicamos um capítulo para cada uma das partes apontadas na Seção 1.2. Apenas o primeiro passo (aquisição dos dados e sincronização do vídeos) não é tratado em um capítulo separado neste texto. Uma vez convertidos e sincronizados os vídeos de um jogo, os índices dos quadros iniciais são gravados em arquivo e assim os vídeos são tratados sempre com o mesmo alinhamento temporal.

O Capítulo 2 mostra um apanhado geral de trabalhos relacionados ao tema deste texto. Tentamos classificar os trabalhos encontrados de forma semelhante à organização dos capítulos para que fosse possível um mapeamento entre as idéias abordadas nos capítulos e os trabalhos relacionados.

No Capítulo 3, apresentamos um método de detecção de jogadores de futebol de salão no plano da quadra usando múltiplas imagens. Esse capítulo representa a primeira fase do trabalho tendo como entrada apenas os vídeos sincronizados e o mapa de pontos característicos descrito na Figura 1.3. Os jogadores são detectados no plano das imagens e os dados de detecção são combinados em uma função multimodal descrita em coordenadas do plano da quadra. Com essa função, podemos encontrar uma estimativa das posições dos jogadores no plano da quadra usando um método de subida pelo gradiente. Ao final do Capítulo 3, apresentamos nossos resultados para a detecção de jogadores (Seção 3.2).

Além de alimentar um processo de estimativa das posições dos jogadores em um dado momento (*frame*) do jogo, a função multimodal também pode ser usada para alimentar a observação de um filtro de partículas. No Capítulo 4, apresentamos uma abordagem de rastreamento dos jogadores em uma partida usando filtros de partículas. Em nosso cenário, os objetos de interesse são os jogadores durante o jogo. Cada jogador possui um estado que o representa em cada momento do jogo. O estado de um jogador é um vetor contendo informações de posição no plano da quadra e uma estimativa de velocidade na direção  $x$  e outra na direção  $y$ . O funcionamento do filtro de partículas se baseia na representação não paramétrica de uma função por meio de um conjunto ponderado de amostras ou partículas. Cada partícula é uma amostra da função que descreve o posicionamento dos jogadores dentro da quadra e o peso da partícula é a probabilidade da partícula ocorrer. O Capítulo 4 mostra o casamento da função multimodal construída no Capítulo 3 com um filtro de partículas. O objetivo é encontrar as trajetórias dos jogadores durante o jogo.

A dinâmica de um jogo faz com que ocorram situações de oclusão, grande proximidade e também colisões. Nesses casos, é preciso uma função de observação capaz de distinguir os jogadores para que o sistema continue funcionando. No Capítulo 5, apresentamos um modelo de aparência que fortalece a observação do filtro de partículas dada pela função multimodal construída no Capítulo 3. O modelo de aparência guarda informações de

aspecto do objeto que está sendo rastreado e serve como ponderador da observação. Com isso, a observação do filtro de partículas recebe baixos valores para partículas que não se referem ao jogador rastreado enquanto que os pesos maiores ficam para as observações corretas.

Ao final de cada capítulo, apresentamos um conjunto de experimentos pertinentes àquele ponto do texto. O Capítulo 5 representa todo o *pipeline* do trabalho. O conjunto de experimentos apresentados na Seção 5.4 representam os experimentos de todas as partes do trabalho juntas, uma vez que representa a última parte do trabalho.

O último capítulo do texto, o Capítulo 6, mostra nossas conclusões e trabalhos futuros.



# Capítulo 2

## Trabalhos Relacionados

O problema tratado em nosso trabalho é o rastreamento visual de jogadores de futsal dentro da quadra de jogo. Como visto no Capítulo 1, nossa abordagem divide o problema em três partes principais:

1. a detecção dos jogadores;
2. o rastreamento dos jogadores e
3. o modelo de aparência.

Esse capítulo busca apresentar os trabalhos relacionados ao tema seguindo uma organização semelhante à organização adotada no conjunto dos capítulos. Alguns trabalhos abordam temas que cobrem mais de uma das partes de nossa abordagem, como é o caso dos trabalhos que tratam de rastreamento e também descrevem a forma de detecção adotada. Nesses casos, preferimos incluir os trabalhos tanto na seção que trata da detecção quanto na seção dedicada aos trabalhos de rastreamento, porém com os comentários adequados à seção.

Muitos outros trabalhos poderiam estar citados aqui considerando que em sua forma mais geral, o tema de rastreamento de múltiplos objetos é bastante abrangente e com inúmeras aplicações. Entretanto, procuramos nos ater aos trabalhos aplicados a cenários próximos de nosso foco específico.

### 2.1 Detecção de Jogadores

A detecção automática de pessoas em imagens digitais, tem sido bastante investigada pela comunidade científica. Aplicações práticas como segurança de ambientes, contagem de pedestres, dispositivos de segurança em automóveis são fortes motivadores dessa linha de pesquisa. O uso de imagens digitais como ferramenta de auxílio em treinamento e

análise de esportes coletivos como futsal também tem despertado o interesse científico. Do ponto de vista da análise, as coordenadas dos jogadores no plano da quadra tem grande importância, principalmente tendo em vista que representam o ponto de partida para a computação de trajetórias, cálculo de velocidades, momentos de grande aceleração, distância percorrida dentre outros que são importantes tanto para o treinamento, quanto em análises de desempenho físico.

No trabalho de Figueroa *et al.* [17] a detecção dos jogadores de futebol é feita por meio da separação do plano de fundo. Por se tratar de um jogo coletivo e praticado ao ar livre, existe uma grande variação de luminosidade, principalmente considerando que grande parte dos jogos realizados no Brasil (local alvo) iniciam ao final do dia e terminam no início da noite. Para lidar com esse problema, os autores fazem a detecção de múltiplos jogadores de futebol por separação de um modelo de fundo semi-estático. Esse modelo é obtido computando uma imagem composta pela média dos *pixels* de um intervalo de *frames* não consecutivos e, portanto, precisa ser calculado periodicamente. Considerando a dinâmica do jogo e câmeras fixas, o resultado da média de um conjunto de *frames* espaçados no tempo resulta em uma imagem contendo uma aproximação do plano de fundo que, quando subtraído de um *frame* do vídeo analisado deixa em destaque os *pixels* dos *blobs*<sup>1</sup> correspondentes aos jogadores. O problema desse tipo de abordagem ocorre quando algum jogador fica estático por um período de tempo suficientemente longo como, por exemplo, um goleiro que assiste o desenrolar do jogo que está concentrado no campo do adversário. Nesse caso, o jogador passa a ser parte do plano de fundo e causa falhas na detecção dos *blobs* envolvidos. Esse problema é contornado modificando o espaço ente os *frames* escolhidos para compor o modelo do plano de fundo. Com isso, um número menor de ocorrências de *pixels* que correspondem ao objeto estático farão parte do cálculo e assim, é menos provável que o problema ocorra.

Outro trabalho com interesse na localização de jogadores de futebol é apresentado por Kang *et al.* [30]. Os autores desse trabalho utilizam vídeos de múltiplas câmeras não calibradas para detectar e rastrear os jogadores. A detecção dos *blobs* é baseada em separação de um modelo de fundo probabilístico. Esse tipo de modelo, assim como o trabalho de Stauffer e Grimson [56], é atualizado a cada novo *frame* da sequência mantendo como modelo de fundo uma imagem onde cada *pixel* possui o valor mais provável de ocorrer. Esse tipo de modelo é bastante robusto a variações de luminosidade mas incorpora *pixels* que representam partes de objetos que interrompem o movimento. O modelo de fundo é subtraído do *frame* a ser analisado para evidenciar os *pixels* que correspondem aos objetos da cena. Se um objeto estaciona na cena, deixará de ser detectado. Se um

---

<sup>1</sup>Os blobs são conjuntos de pontos que correspondem a um mesmo objeto de interesse e podem ser encontrados de diversas formas, dependendo da natureza do problema. Por exemplo, se o objeto de interesse é formado de *pixels* de mesma cor, então os blobs que correspondem a objetos de interesse serão compostos por *pixels* que possuem a cor correspondente.

objeto estacionado voltar a se mover passa a ser detectado novamente mas também deixa para trás uma nova detecção indesejada na posição onde estava parado.

O trabalho apresentado por Tong *et al.* [58] combina subtração de plano de fundo baseado em cores dominantes e uma detecção formada por agrupamentos de filtros de Haar. O sistema apresentado pelos autores encontra pontos característicos usando *Hough lines*. Esses pontos característicos são usados para permitir um mapeamento entre as posições encontradas no plano das imagens e as posições correspondentes no plano do campo de jogo. Os vídeos de entrada são jogos de futebol usados em transmissão de televisão e portanto bastante dinâmicos. Por outro lado, as transmissões de televisão tornam impossível a visualização de todos os jogadores simultaneamente.

Ming *et al.* [37] apresentam um trabalho cujo objetivo é a detecção de jogadores de futebol. O método utiliza um modelo de fundo estatístico e probabilístico construído usando mistura de Gaussianas, tal como no trabalho de Stauffer e Grimson [56]. Esse tipo de modelo para o plano de fundo mostra robustez a variações de luminosidade mas incorpora ao modelo *pixels* correspondentes a objetos que se tornam estáticos na cena. A principal diferença entre os dois trabalhos está em um pré-processamento que elimina as partes da imagem que não correspondem a regiões do campo de jogo.

Uma abordagem genérica para detectar pessoas usando um conjunto fixo de câmeras omnidirecionais<sup>2</sup> é apresentado por Alahi *et al.* [2]. Apesar de se tratar de um método genérico para detecção de pessoas, os autores direcionaram o trabalho para observar jogos de basquetebol. O modelo usa as silhuetas dos jogadores como observação e enfrenta problemas como o brilho do polimento da quadra e reflexos dos jogadores.

Khan e Shah [31] desenvolveram uma técnica para localização de pessoas baseada em homografia e múltiplas câmeras. O modelo de fundo é calculado utilizando a técnica adaptativa de Stauffer e Grimson [56], baseado em mistura de Gaussianas. O resultado da separação do plano de fundo de Khan e Shah é um mapa de probabilidades onde cada *pixel* guarda a probabilidade de ser parte do *foreground*. Usando homografia, os pontos dos mapas resultantes em cada câmera são projetados no plano da imagem de uma das câmeras, chamada de imagem referência. As probabilidades dos mapas gerados em câmeras diferentes são multiplicadas fundindo os dados em um mapa geral com picos nas regiões de intercessão. Os picos são segmentados por limiarização e representam as regiões com maior probabilidade de se encontrar uma pessoa. Esse método assume que a região de interesse está no campo de visão de todas as câmeras envolvidas, caso contrário, não conseguiria fazer a segmentação final por limiarização. Nas regiões que não são cobertas por todas as câmeras ao mesmo tempo, os picos encontrados não se adequam

---

<sup>2</sup>Um sistema omnidirecional permite a aquisição de imagens com um campo de visão de 360°. O sistema pode ser obtido através da combinação das imagens de apenas uma câmera que gira em torno do próprio eixo, combinação de imagens de oriundas de múltiplas câmeras ou a utilização de espelhos [64].

ao limiar global e, portanto, não podem ser considerados como objetos detectados.

Diferente das abordagens anteriores, podemos tratar o problema de encontrar jogadores no plano da imagem como um problema de detecção de objetos de interesse em sua forma geral. Em nosso caso, o objeto de interesse é um jogador dentro de quadra de jogo. No ano de 2001, Viola e Jones [61] propuseram um detector de objetos com funcionamento em tempo real para pequenas imagens. Apesar do foco do trabalho ser direcionado para faces, o método é perfeitamente adaptável para a detecção de outros objetos de interesse como pessoas ou ainda jogadores de futsal durante o jogo. O que determina o objeto de busca é um passo de treinamento supervisionado que antecede a detecção. A etapa de treinamento conta com uma coleção composta por exemplos de objetos de interesse e outra coleção composta por exemplos do plano de fundo. Durante o treinamento, o algoritmo de AdaBoost [21] é usado para organizar uma cascata de classificadores do tipo Haar [45] de tal forma que os classificadores de menor complexidade ficam no início da cascata. Assim, durante a detecção, as janelas de observação que não contêm o objeto de interesse são descartadas logo nos primeiros estágios da cascata, liberando processamento para as tentativas com maior probabilidade de representarem uma detecção correta. O resultado do método é um conjunto de retângulos em coordenadas da imagem, cada um delimitando um recorte que contém uma detecção.

Recentemente, o trabalho de Felzenszwalb *et al.* [14] apresenta um método para detecção de objetos baseado em mistura de modelos de partes deformáveis em multi-escala para representar objetos com grande variabilidade. O método é treinado usando um procedimento discriminativo que necessita apenas de um conjunto de imagens contendo marcações retangulares. O sistema resultante é considerado eficiente e preciso, conseguindo bons resultados avaliando coleções difíceis como PASCAL VOC benchmarks e INRIA Person. O sistema representa objetos por uma coleção de partes arranjadas em uma configuração deformável. Cada parte captura propriedades locais de aparência de um objeto enquanto que a configuração deformável conecta partes aos pares.

Além de detectar os jogadores nas imagens, nosso trabalho precisa rastrear os múltiplos objetos detectados para determinar uma aproximação de sua trajetória no plano da quadra de jogo. A próxima seção mostra trabalhos relacionados ao rastreamento dos jogadores.

## 2.2 Rastreamento de Jogadores

Nos últimos anos, alguns trabalhos tem voltado sua atenção à detecção e rastreamento de jogadores de esportes coletivos. Esse tipo de investigação possui aplicações diversas não apenas no meio esportivo mas também em ambientes diferentes mas com características semelhantes. Monitorar o fluxo de pessoas em corredores de grandes lojas para avaliar a distribuição dos produtos ou verificar a eficácia das vitrines, podem ser aplicações co-

merciais que envolvem o rastreamento de múltiplos objetos simultâneos. Aplicações em segurança também podem ser contempladas usando rastreamento de múltiplos indivíduos, tanto em ambientes internos quanto externos. São muitas as aplicações que podem ser beneficiadas usando rastreamento visual de objetos, mas o nosso foco neste trabalho é o rastreamento de jogadores de futsal.

Em seu trabalho, Figueroa *et al.* [17, 18] adotam uma abordagem interessante para encontrar as trajetórias dos jogadores de Futebol. O método proposto transforma o problema de rastreamento visual em um problema de menor caminho em um grafo. No grafo, os nós representam os *blobs* detectados durante a análise dos vídeos de um jogo. Cada aresta liga um *blob* a outro *blob* do próximo quadro da sequência e os pesos das arestas equivalem a distâncias. Assim, cada *blob* está conectado a todos os outros do próximo quadro da sequência por meio de arestas ponderadas. Determinando no primeiro quadro qual jogador deve ser rastreado ou qual nó será o ponto de partida, o método encontra o menor caminho no grafo até um nó final e faz o mapeamento para coordenadas no plano do jogo.

Técnicas baseadas em silhuetas também tem sido utilizadas. No trabalho de Alahi *et al.* [2], encontramos um método baseado em silhuetas que detecta jogadores de basquetebol e faz o rastreamento baseado na menor distância entre detecções de quadros adjacentes.

O método apresentado por Kang *et al.* [30] também é voltado para jogadores de futsal. Os autores propõem um método de rastreamento de jogadores usando vídeos de múltiplas câmeras e uma abordagem de detecção de *blobs*. A detecção dos *blobs* é baseada na separação de um modelo de fundo probabilístico dos objetos de interesse. Os *blobs* detectados são projetados em um plano virtual que corresponde ao plano da quadra de jogo e o rastreamento usa filtro de Kalman para modelar o movimento.

Outros trabalhos utilizam múltiplos pontos de vista para tratar oclusões. No trabalho de Miura e Kubo [39], os autores encontram iterações entre a bola e os jogadores durante uma partida de futebol. O campo é observado por três pares de câmeras, onde cada par observa a mesma cena de lados opostos do campo. Cada câmera possui um nó de processamento associado que se comunica com um nó supervisor. O nó supervisor cria um ambiente virtual com os dados vindos dos nós inferiores para encontrar as iterações dos jogadores com a bola. Outra abordagem utilizando múltiplas câmeras é mostrada em Inamoto e Saito [25]. Nesse trabalho, a bola e os jogadores são segmentados por subtração de fundo, as sombras são retiradas por análise de cores e, finalmente, as correspondências dos jogadores ao longo dos quadros do filme são feitas por homografia.

Alguns autores utilizam técnicas probabilísticas para estimar o novo estado dos objetos de interesse. Esse tipo de técnica utiliza um conjunto de equações conhecido com dinâmica para modelar o movimento dos objetos de interesse. Alguns trabalhos utilizam técnicas

baseadas no filtro de Kalman para modelar a dinâmica dos objetos como, por exemplo, os trabalhos de Ababsa e Mallem [1] e Merhi *et al.* [36]. O filtro de Kalman [23, 62] consiste em um conjunto de equações matemáticas que fornecem uma maneira computacional para estimar os estados de um processo, minimizando a média dos erros quadrados [36]. Esse é um filtro poderoso com características interessantes como a capacidade de estimar estados passados e futuros. Entretanto, as técnicas baseadas em filtros de Kalman trabalham bem com processos Gaussianos e, conseqüentemente, não conseguem lidar bem com processos multimodais ou multi-hipótese, como é o caso de dois objetos com trajetórias que se cruzam durante uma cena. Uma possível solução é o uso de técnicas baseadas em filtros de partículas [23, 26].

Filtro de partículas [23, 26] é um método seqüencial de Monte Carlo baseado na representação de densidades de probabilidades por pontos de massa, chamados de partículas [1]. Cada partícula é uma amostra da função que está sendo representada, ou uma instância de um possível estado do processo que está sendo simulado. Essa forma não paramétrica de representação dá ao filtro de partículas a habilidade de modelar funções não lineares e não Gaussianas e, portanto, o torna capaz de lidar com múltiplas hipóteses. O filtro de partículas é um filtro preditivo capaz de estimar o novo estado de um objeto rastreado por meio de uma função de distribuição de probabilidades *a posteriori*, representada de forma não paramétrica, e uma função de predição baseada em uma dinâmica de movimento conhecida [23]. Alguns trabalhos que lidam com rastreamento visual de objetos que se movem em uma cena são baseados nesta técnica, como é o caso de Freitas de Moraes [20], Isard e Blake [26], Moraes *et al.* [43], Singh *et al.* [54] e Ababsa e Mallem [1].

O problema de rastrear jogadores em uma partida de futsal lida com objetos de interesse que possuem dinâmica de movimento bastante ativa, o que torna o problema ainda mais difícil. No entanto, soluções para esse problema têm aplicações em diferentes campos da ciência. Em um sistema de rastreamento visual, os objetos de interesse precisam de uma representação que permita sua detecção a cada novo quadro da seqüência, conhecido como modelo de observação [65]. Normalmente, o conhecimento sobre sua aparência permite sua detecção e, conseqüentemente, a sua representação no sistema de rastreamento.

A terceira parte de nosso trabalho trata do modelo de aparência usado durante a detecção dos objetos de interesse. Em nosso caso, o modelo de aparência é usado para fortalecer as medições do filtro de partículas usado no rastreamento dos jogadores de futsal. A próxima seção apresenta alguns trabalhos relacionados a este tópico de interesse.

## 2.3 Modelo de aparência

Em um sistema de rastreamento, o interesse principal é monitorar e encontrar correspondências de um dado de interesse ao longo do tempo observando imagens. Em nosso

contexto, estamos interessados em rastrear objetos de interesse em uma cena conhecida. Para isso, o objeto de interesse precisa ser modelado de tal forma que possa ser observado pelo rastreador e conseqüentemente monitorado através do *frames* de um vídeo.

De acordo com Yilmaz *et al.* [65], um objeto pode ser representado por sua forma e por sua aparência. Em geral, objetos que ocupam pouco espaço nas imagens são representados por um ponto central [60] ou por um conjunto de pontos [51]. No trabalho de Serby e Gool [51], encontramos um método voltado ao rastreamento de objetos de forma genérica, levando em conta que consegue manipular características de objetos diversos e agrupa várias características diferentes para obter uma representação melhor do objeto rastreado. Como inicialização do método, um usuário precisa delinear o objeto de interesse para que o método possa extrair as características utilizadas e dar início ao rastreamento. As características extraídas pelo método são integradas no processo de observação de um filtro de partículas para o rastreamento do objeto observado. Outros trabalhos também usam características para representar seus objetos de interesse. Isard e Blake [27] combinam cores e informações a respeito do contorno para representar os objetos rastreados.

Alguns trabalhos representam os objetos observados usando formas primitivas como retângulos ou elipses. Em [20], os objetos de interesse são representados por elipses delimitadoras. Por meio de uma separação do plano de fundo usando um modelo probabilístico, um mapa de probabilidades para o *foreground* é obtido e usado como entrada no passo de observação de um filtro de partículas. As elipses são ponderadas observando as probabilidades dos *pixels* em seu interior constituírem o *foreground*. Quanto mais *pixels* de *foreground* maior será o peso final da elipse.

Silhuetas e contornos também são usados como representação de objetos, normalmente usando formas complexas não rígidas [66]. Alahi *et al.* [2] apresentam um trabalho com detecção e rastreamento focado em jogadores de basquete. Nesse trabalho, os autores utilizam silhuetas como modelo do objeto de interesse observado em um conjunto fixo de câmeras omnidirecionais.

Um modelo comumente usado para o reconhecimento de objetos tenta encontrar uma representação do esqueleto aplicando operações e análises à silhueta dos objetos de interesse [3, 5]. Outra representação divide o objeto em partes conectadas por articulações, tal como o corpo humano. Uma pessoa sendo rastreada pode ser representada por partes simples que correspondem às pernas, os braços, à cabeça e ao corpo. Quando o objeto modelado está em movimento, as partes se movem de forma diferente e o movimento precisa ser modelado considerando as diferentes partes. Para cada uma das partes, é associada uma dinâmica de movimento que depende diretamente do objeto observado [65].

Os objetos de interesse também podem ser representados por sua aparência. Modelos de aparência são usados normalmente para encontrar o objeto nas imagens. Considerando que esse tipo de representação indica o aspecto que o objeto de interesse deve ter nas

imagens analisadas, a representação por aparência passa a ser uma ferramenta interessante para verificar a existência e, portanto, a localização, de objetos de interesse nas imagens. Isso caracteriza a detecção de objetos.

Uma forma de representar aparência de objetos é por meio da densidade de probabilidade da aparência do objeto. Esse tipo de representação utiliza conceitos de probabilidade para representar matematicamente uma certa característica do objeto, como a distribuição de suas cores. A função pode ser paramétrica como uma Gaussiana [67], uma mistura de Gaussianas [46], ou pode ser não paramétrica como um histograma de cores [9]. Uma representação por meio de função de densidade de probabilidade pode ser obtida analisando a região no interior de uma silhueta, por exemplo.

Os *templates* são formados usando formas simples como retângulos ou as formas complexas como silhuetas dos objetos [15]. O uso de *templates* simplifica a representação do objeto. Entretanto, possui limitações que impedem o seu uso em várias situações. Um *template* é fixo e não consegue representar mudanças no objeto como a mudança de pose ou mudança do ponto de visualização. Essas são mudanças que podem ocorrer no objeto observado e que impedem o uso de uma estrutura fixa como um *template*. Seu uso ocorre normalmente em situações onde existe apenas um ponto de observação e a pose do objeto não varia significativamente.

Os modelos de aparência ativos (*Active Shape Models - ASM*) são formados por uma combinação da representação da forma com a representação de aparência [13]. Esse tipo de modelo necessita de uma fase de treinamento para que sua parte relativa à forma e também a parte relativa à aparência possam ser aprendidas. Para isso, são necessários exemplos do objeto de interesse. Em geral, a forma do objeto é representada por marcações, tal como na representação de contornos onde as marcas são definidas nos limites do objeto. Para cada marca, é associada uma aparência na forma de vetor de cores, textura ou magnitude de gradiente [65].

Modelos baseados em múltiplas visões do objeto codificam diferentes visões simultaneamente. Uma maneira de fazer esse tipo de codificação é a construção de sub-espacos das diferentes visões, por exemplo, usando Análise de Componentes Principais (PCA) ou Análise de Componentes Independentes (ICA) [8, 40, 65]. Outra forma de detectar objetos em diferentes pontos de observação é por meio de treinamento de conjuntos de classificadores, como *Support Vector Machines* [4] ou Redes Bayesianas [47]. Uma limitação desse tipo de representação está no fato de todas as visões serem necessárias para a detecção correta.

Em nosso trabalho, utilizamos uma estrutura baseada em três etapas como visto no Capítulo 1. A primeira trata da localização do jogadores de Futebol de Salão no plano da quadra. Para encontrar as posições dos jogadores, usamos um método de detecção de objetos clássico em Visão Computacional baseado em aprendizado de máquina. Esse

método de detecção precisa de treinamento prévio e, para isso, separamos uma coleção de exemplos marcados manualmente. Os dados encontrados nas imagens são projetados em um plano virtual que corresponde ao plano da quadra de jogo. Os dados projetados são fundidos em uma função que dá a probabilidade de uma posição da quadra conter um jogador. A segunda apresenta o método de rastreamento adotado para recuperar as trajetórias. Em nosso caso, usamos a função de probabilidades encontrada no passo anterior como observação em um filtro de partículas para fazer o rastreamento. A terceira mostra o modelo de aparência adotado para fortalecer a observação do sistema de rastreamento. Usamos um conjunto de histogramas como modelo de aparência. Os histogramas possuem informações de cor e forma dos objetos de interesse. Os próximos capítulos mostram maiores detalhes de cada uma das etapas.



# Capítulo 3

## Localização no Plano da Quadra

Neste Capítulo, estamos interessados em encontrar uma estimativa das posições dos jogadores no plano do jogo, a cada instante da partida. Esse resultado será útil para o cálculo das trajetórias percorridas pelos jogadores durante o jogo e, conseqüentemente, possibilitará o cálculo de outras informações de grande importância para as equipes técnicas.

Como descrito no Capítulo 1, o cenário consiste de um conjunto de quatro câmeras posicionadas ao redor da quadra. Este posicionamento garante que qualquer área da quadra está enquadrada por pelo menos duas câmeras do conjunto. Em nosso estudo, todos os jogos foram gravados usando esta configuração. Entretanto, o método aqui descrito, pode ser adaptado para uma configuração diferente. Com mais câmeras observando a mesma cena, diminui-se a probabilidade de um objeto não ser observado, já que, uma oclusão na visão de uma câmera pode não ocorrer em outra. Logo, um objeto perdido em uma das câmeras pode ser detectado na visão de outra câmera.

Para tratar o nosso problema, os únicos dados de entrada são os vídeos registrados pelas câmeras e um mapa da quadra com coordenadas de pontos estratégicos como, por exemplo, a marca de pênalti, o centro e os cantos da quadra. As coordenadas contidas no mapa servirão para o cálculo de uma matriz  $H$  conhecida como matriz de homografia. Com o uso dessa operação projetiva, conseguimos mapear coordenadas no plano da imagem em coordenadas no plano da quadra. Portanto, a cada câmera é preciso associar uma matriz  $H$  individual e fixa, já que as câmeras são estacionárias.

O estágio seguinte consiste em encontrar os objetos de interesse (no caso os jogadores) no plano da imagem em cada câmera. Para isso, podemos usar um detector de objetos que esteja treinado para localizar pessoas no plano das imagens. Opções interessantes são o método proposto por Viola e Jones [61] e a técnica de Felzenszwalb *et al.* [14]. Nos dois casos, as técnicas podem ser treinadas para detectar pessoas e os resultados são dados em forma de retângulos que envolvem o objeto detectado na imagem.

Como observação no plano da imagem, estamos interessados na localização dos pés

dos jogadores. Podemos assumir que o ponto médio da base de um retângulo detectado representa bem a localização do ponto inferior de um jogador na imagem. Portanto, nossa observação será constituída pelo conjunto das localizações destes pontos no plano das imagens, ou o conjunto das bases médias dos retângulos encontrados pelos detectores.

Nossa técnica projeta a observação de múltiplas câmeras em um mundo virtual correspondente ao plano da quadra para fundí-las em uma função bidirecional e multimodal. O domínio dessa função é o conjunto de coordenadas  $(x, y)$  no plano da quadra e a imagem da função será a probabilidade de um jogador ser encontrado em tal ponto. No processo de fusão, cada ponto projetado é transformado em uma Gaussiana e as posições dos jogadores no plano da quadra são encontradas pela subida do gradiente.

As próximas seções mostram os detalhes usados para a localização multicâmera, seguindo a ordem: A Seção 3.1 descreve os estágios separadamente, passando pela observação (Seção 3.1.1), projeção (Seção 3.1.2), e localização (Seção 3.1.3). Em seguida, apresentamos uma seção de resultados (Seção 3.2) que compara os dados obtidos utilizando a fusão multicâmera e localizações usando um detector clássico em visão computacional com o maior número de citações nos últimos dez anos.

## 3.1 Descrição da Técnica

Nesta seção, apresentamos em maiores detalhes os estágios da técnica multicâmera para localização de jogadores. A técnica se resume em quatro estágios bem definidos, como mostra a Figura 3.1. Cada um dos estágios é mostrado em um dos tópicos seguintes que mantém a mesma ordem mostrada na figura.

### 3.1.1 Detecção Independente no Plano da Imagem

Alguns trabalhos utilizam informações de cor para detectar os objetos de interesse, como em Juang *et al.* [29]. Outros fazem extração de fundo para encontrar os *blobs* que correspondem ao objeto de interesse, como em Miura e Kubo [39] e Figueroa *et al.* [18]. Em nossa abordagem, a localização automática de jogadores de futebol de salão baseada em fusão de observações multicâmera precisa de um método para detectar os jogadores no plano das imagens e construir a observação. Em nosso trabalho, selecionamos o método de detecção de objetos proposto por Viola e Jones [61]. Esse é um método clássico de fácil utilização e que promete bons resultados, além de muito citado na literatura de visão computacional e possuir uma fase de treinamento simples, apesar de onerosa.

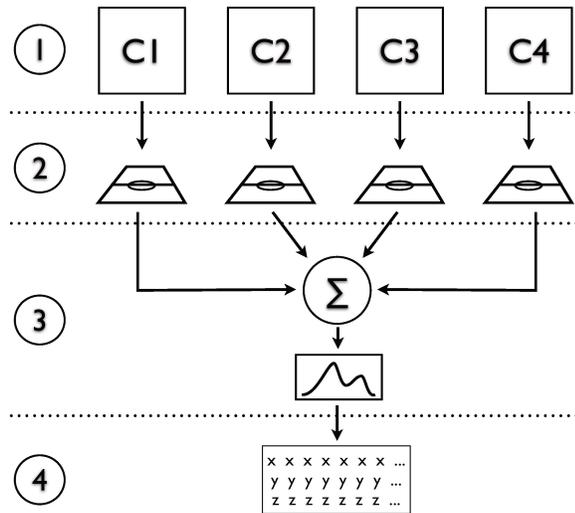


Figura 3.1: Diagrama com os estágios principais do modelo proposto para localização de jogadores. No Estágio 1, os jogadores são detectados no plano da imagem usando um detector treinado para encontrar jogadores de futebol de salão em cada uma das câmeras. No Estágio 2, as observações são projetadas no plano da quadra usando homografia. No Estágio 3, os dados projetados são fundidos em uma função de probabilidade bidimensional e multimodal para a localização dos jogadores no plano da quadra. No Estágio 4, as posições dos jogadores são encontradas usando a subida pelo gradiente.

### O método de Viola e Jones

O trabalho desenvolvido por Viola e Jones [61] descreve um método baseado em aprendizado de máquina para detecção visual de objetos com foco em detecção de faces, capaz de processar imagens rapidamente e mantendo taxas elevadas de detecção. Além disso, apesar de a aplicação inicial proposta ser a detecção de faces, o método pode ser treinado para detectar outros objetos de interesse como, por exemplo, pessoas.

O detector de faces proposto por Viola e Jones trabalha com características bem simples, comumente denominadas *Haar-like features*. Uma característica é calculada como a diferença de retângulos adjacentes como mostram os exemplos da Figura 3.2.

Uma característica de dois retângulos consiste na diferença entre a soma das intensidades dos *pixels* encontrados em um retângulo pela soma das intensidades dos *pixels* encontrados no retângulo adjacente, mostrados na Figura 3.2 com cores diferentes. Em uma característica, os dois retângulos possuem as mesmas dimensões e são adjacentes verticalmente ou horizontalmente. Uma característica de três retângulos considera três retângulos semelhantes e adjacentes. Depois de calcular o somatório dos *pixels* de cada região, o valor da característica é encontrado somando-se os valores encontrados nos retângulos laterais

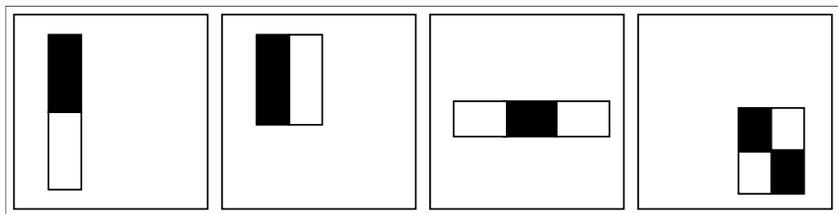


Figura 3.2: Exemplos de características do tipo dois, três e quatro retângulos. O valor de uma característica é dado pela diferença das somas de regiões marcadas em cores diferentes. No caso da figura, o valor de uma característica será a diferença entre as partes pretas e as partes brancas.

e depois, subtraindo-se o total encontrado no retângulo que está no meio. Por último, a característica de quatro retângulos computa a diferença entre os pares de retângulos em diagonal.

O método utiliza como imagem base para a localização, janelas deslizantes, que originalmente possuem  $24 \times 24$  *pixels* para detectar faces. Dentro de cada uma dessas janelas, existem mais de 180.000 características com tamanhos e orientações diferentes. Para permitir que o cálculo das somas seja rápido, o método não utiliza as intensidades dos *pixels* diretamente, mas sim valores encontrados em uma imagem chamada de *imagem integral* [61]. Uma *imagem integral* é uma imagem com as mesmas dimensões da imagem original em que cada ponto possui o valor da soma das intensidades de todos os *pixels* originais acima e à esquerda do ponto.

$$ii(x, y) = \sum_{x' \leq x, y' \leq y} i(x', y'). \quad (3.1)$$

A imagem integral (*ii*) pode ser computada com apenas uma leitura em cada *pixel* da imagem consumindo pouco tempo de processamento. Com ela, é possível encontrar o somatório de um retângulo computando apenas três somas de elementos da imagem integral como mostra a Figura 3.3.

Uma característica do tipo *Haar* constitui um classificador fraco [45]. Um grande número de características é necessário para se ter uma precisão considerável. Para isso, o método de Viola e Jones organiza uma sequência de classificadores em cascata, onde a saída de um se liga à entrada do outro, formando um classificador forte (Figura 3.4).

A cascata de classificadores é usada para alcançar uma taxa de detecção elevada. Os classificadores são organizados de forma que os custos computacionais vão crescendo ao longo da cascata. A idéia é eliminar rapidamente, nos primeiros estágios, o maior número possível de tentativas que não representam um objeto de interesse e, com isso, deixar apenas as janelas relevantes para os estágios finais da cascata. Uma janela analisada pelo método só será considerada como contendo um objeto de interesse se for aceita por todos

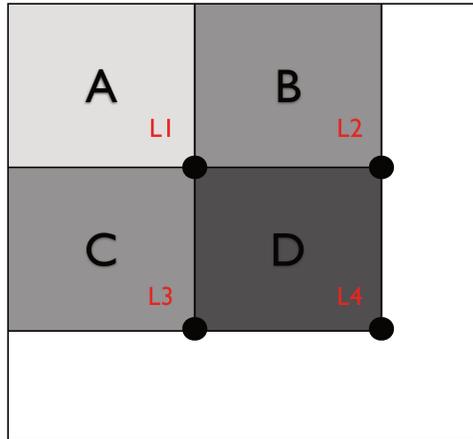


Figura 3.3: Exemplo de Imagem Integral *ii*. A soma dos pontos em **A** é dada pelo valor de  $L1$ . A soma dos pontos em **B** é dada por  $L1 - L2$ , a soma em **C** é dada por  $L3 - L1$  e a soma em **D** é dada por  $L4 - L3 - L2 + L1$ .

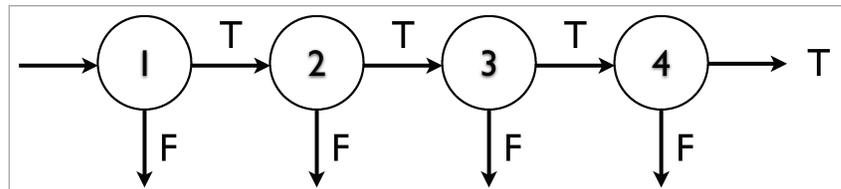


Figura 3.4: Cascata de classificadores do tipo Haar. Uma subimagem é submetida à cascata. Cada classificador fraco avalia a janela para verificar se corresponde a um objeto de interesse. Em caso negativo, a busca é interrompida. Uma subimagem contém um objeto de interesse se passar pela avaliação de todos os classificadores da cascata.

os estágios da cascata de classificadores.

A técnica Adaboost [21] é usada para selecionar as melhores características para construir o classificador em cascata e organizá-los na sequência mais adequada. Para cada classificador fraco, o treinamento determina o melhor limiar que resulta no menor número de exemplos classificados erroneamente, tanto para a classe de objeto quanto de não objeto. A construção do classificador forte no final consiste na combinação dos classificadores fracos em sequência.

### 3.1.2 Projeção da Localização em Coordenadas de Imagem no Plano da Quadra

As observações dos detectores estão descritas em coordenadas no plano das imagens. Entretanto, estamos interessados em localizar os jogadores no plano da quadra. Vários pontos característicos da quadra possuem coordenadas conhecidas de antemão. Com isso,

é possível usar um conjunto de correspondências para encontrar uma matriz  $H$  que mapeia pontos de um plano  $a$  a seus correspondentes em um plano  $b$ . Essa matriz é conhecida como matriz de homografia. Usando a quadra como o plano  $a$ , precisamos encontrar as correspondências dos pontos conhecidos descritas em um plano  $b$ , que no caso é o plano da imagem. Essas correspondências podem ser obtidas usando um quadro do vídeo capturado para marcar as correspondências entre pontos.

A operação de homografia é uma transformação projetiva que mapeia coordenadas de um plano para coordenadas de outro plano. Normalmente, a homografia é usada para encontrar correspondências de pontos entre imagens da mesma cena. A homografia descreve o que ocorre quando o ponto de observação de uma cena é modificado. É possível extrair a rotação e translação da câmera de uma matriz de homografia que foi estimada, com o propósito de inserir objetos virtuais baseados em modelos 3D de forma a parecerem ser parte da cena.

Podemos considerar uma cena qualquer sendo observada por uma câmera  $a$  e um ponto  $p_a = (x, y, 1)$  resultante da projeção de um ponto  $p$  da cena em  $a$  e descrito em coordenadas homogêneas. A mesma cena também é observada por outra câmera  $b$  e portanto existem correspondências de pontos nas duas câmeras. Se o ponto  $p$  for observado pelas duas câmeras, então:

$$p_b = H_{ab}p_a, \quad (3.2)$$

ou seja, o ponto  $p_b$  é igual ao ponto  $p_a$  modificado pela homografia  $H_{ab}$ , que é capaz de transformar um ponto do plano  $a$  em um ponto no plano  $b$ .

A homografia  $H$  é descrita com uma matriz  $3 \times 3$  inversível, portanto:

$$H_{ba} = H_{ab}^{-1} \quad (3.3)$$

$$p_a = H_{ab}^{-1}p_b = H_{ba}p_b. \quad (3.4)$$

Para estimar a homografia  $H$ , precisamos partir da Equação 3.2 escrita como:

$$\begin{bmatrix} x_{p_b} \\ y_{p_b} \\ z_{p_b} \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} = \begin{bmatrix} x_{p_a} \\ y_{p_a} \\ z_{p_a} \end{bmatrix} \Leftrightarrow p_b = Hp_a. \quad (3.5)$$

Em coordenadas não homogêneas  $x'_{p_a} = x_{p_a}/z_{p_a}$  e  $y'_{p_a} = y_{p_a}/z_{p_a}$ ,

$$x'_{p_a} = \frac{h_{11}x_{p_a} + h_{12}y_{p_a} + h_{13}z_{p_a}}{h_{31}x_{p_a} + h_{32}y_{p_a} + h_{33}z_{p_a}}, \quad (3.6)$$

$$y'_{p_a} = \frac{h_{21}x_{p_a} + h_{22}y_{p_a} + h_{23}z_{p_a}}{h_{31}x_{p_a} + h_{32}y_{p_a} + h_{33}z_{p_a}}. \quad (3.7)$$

Sem perder generalidade, podemos fazer  $z_{p_a} = 1$  e rearranjar a equação até obter

$$a_x^T h = 0 \quad (3.8)$$

$$a_y^T h = 0, \quad (3.9)$$

onde

$$\begin{aligned} h &= (h_{11}, h_{12}, h_{13}, h_{21}, h_{22}, h_{23}, h_{31}, h_{32}, h_{33})^T \\ a_x &= (-x_{p_a}, -y_{p_a}, -1, 0, 0, 0, x'_{p_b} x_{p_a}, x'_{p_b} y_{p_a}, x'_{p_b})^T \\ a_y &= (0, 0, 0, -x_{p_a}, -y_{p_a}, -1, y'_{p_b} x_{p_a}, y'_{p_b} y_{p_a}, y'_{p_b})^T. \end{aligned}$$

Com um conjunto de pontos correspondentes entre os planos, é possível resolver o sistema linear

$$Ah = 0, \quad (3.10)$$

onde

$$A = \begin{pmatrix} a_{x_1}^T \\ a_{y_1}^T \\ \vdots \\ a_{x_N}^T \\ a_{y_N}^T \end{pmatrix}. \quad (3.11)$$

A homografia mapeia coordenadas de um plano projetadas em outro plano. Em nosso caso, os objetos de interesse se movem sobre o plano da quadra de jogo. Podemos usar a homografia de pontos específicos, como os pés dos jogadores para encontrar suas localizações no plano da quadra. Cada jogador encontrado pelo detector é representado por um retângulo no plano da imagem.

Em nosso trabalho, consideramos a base média desse retângulo como uma boa representação da localização do ponto inferior do jogador no plano da imagem. Considerando que a estimativa da posição do pé de um jogador no plano da imagem não será exata, a projeção para o plano da quadra dada pela homografia não terá como resultado o ponto exato de onde está o jogador. Por outro lado, a projeção de todos os pontos encontrados nas imagens resulta em um conjunto importante de indícios das localizações.

O resultado da detecção em cada uma das imagens tem como resultado uma lista de retângulos e, portanto, uma lista de pontos que representam os pés dos jogadores detectados nas respectivas imagens. Esses pontos são projetados no plano da quadra usando as respectivas matrizes de homografia. A nova coleção resultante representa as correspondências dos pontos encontrados pelos detectores descritas em coordenadas do mundo.

Após a fusão dos dados encontrados nas múltiplas câmeras, obtemos uma função de probabilidades multimodal que retorna a probabilidade de ser encontrado um jogador em cada ponto no plano da quadra. Essa função será usada para a localização dos jogadores no plano da quadra. A próxima seção mostra como é construída a função de probabilidades para encontrar o potencial de um jogador estar em uma dada posição da quadra.

### 3.1.3 Fusão dos Dados Multi-Câmera

Após a detecção dos jogadores no plano de cada uma das múltiplas câmeras, tem-se um conjunto de observações dos pés dos jogadores em cada câmera. Cada retângulo detectado representa a localização de um jogador na câmera de origem. Estamos assumindo que o ponto médio da base de cada retângulo é uma boa estimativa da localização do pé do jogador na imagem. Projetamos esse ponto no plano da quadra usando a matriz de homografia correspondente e, com isso, temos um novo conjunto de pontos, agora em coordenadas do plano da quadra e composto pela projeção das observações nas diferentes câmeras.

Sabemos que esses pontos não correspondem à localização real dos jogadores, já que existem erros acumulados desde sua detecção até sua projeção no plano da quadra. Entretanto, os pontos projetados no plano da quadra representam fortes evidências da presença de um jogador naquela região. Podemos representar essa região no plano da quadra usando uma função Gaussiana com parâmetros que variam de acordo com a câmera que deu origem à observação correspondente. Os parâmetros (médias e covariâncias) destas funções Gaussianas em cada câmera, são encontrados usando vídeos destinados ao treinamento do sistema.

Os vídeos de treinamento possuem anotação das posições dos jogadores em coordenadas da quadra em cada quadro, para que essas posições possam ser comparadas às posições detectadas e projetadas no plano da quadra. Comparando os pontos projetados por uma câmera com os pontos marcados, conseguimos calcular o erro médio e a covariância do erro de projeção da câmera e determinar a função Gaussiana que representa esse erro. Para isso, é preciso fazer a correspondência entre os pontos marcados e os pontos detectados. Estamos considerando o ponto projetado mais próximo como sendo o correspondente de um ponto marcado, mas em casos de vários jogadores próximos podem haver confusões. Para evitar esse tipo de erro, no treinamento foram consideradas apenas as correspondências que não geram dúvidas, ou seja, situações em que a marcação mais próxima está relativamente perto do ponto projetado e a segunda marcação mais próxima está relativamente longe. Para escolher esses pontos foram usados dois limiares: escolhemos as correspondências em que a marcação mais próxima de um ponto projetado está a menos de dois metros ( $L_1$ ) e a segunda marcação mais próxima está a mais de três metros ( $L_2$ ).

Dadas as correspondências entre os pontos marcados nos vídeos de treinamento e as projeções de cada câmera, podemos computar a média, o erro de projeção nas direções  $x$  e  $y$  e também a matriz de covariância do erro de projeção para cada uma das câmeras. Assim, podemos dizer que cada câmera possui um erro de projeção sistemático medido pela média dos erros e uma matriz de covariância que representa a incerteza na projeção, ambos medidos analisando os vídeos de treinamento.

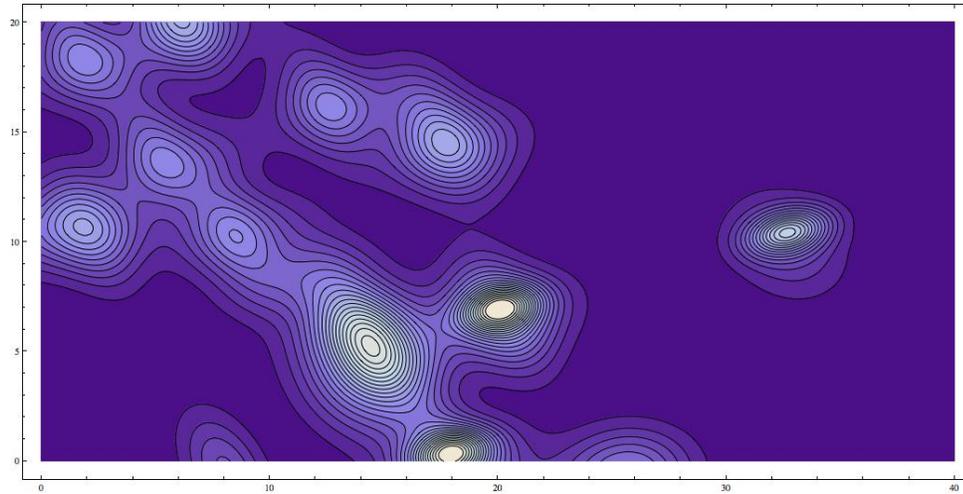


Figura 3.5: Função de probabilidade 2D. As projeções multicâmera são substituídas por funções Gaussianas com parâmetros treinados. Como resultado, tem-se uma função 2D que retorna a probabilidade de se encontrar um jogador em uma dada região. Na figura, um exemplo dessa função é representada em curvas de nível.

Conhecidas as médias e covariâncias de cada câmera, podemos observar que ao redor de um ponto projetado existe uma região na qual poderemos encontrar o jogador correspondente, ou que cada ponto projetado consiste na localização de um jogador acrescida de um erro. Dessa forma, cada ponto projetado dá lugar a uma função Gaussiana 2D centrada na posição do ponto projetado, corrigida pelo erro médio da projeção e com a covariância calculada para a câmera correspondente, tudo determinado na fase de treinamento. A combinação das funções Gaussianas de cada câmera resulta em uma função única para toda a quadra capaz de retornar a probabilidade de um jogador ser encontrado em uma dada região da quadra. Um exemplo dessa função é exibido na Figura 3.5.

Durante os trechos de análise, uma função de probabilidade é formada a cada novo *frame* do vídeo com base nas projeções das múltiplas câmeras. As posições de maior probabilidade correspondem às posições mais prováveis de se encontrar um jogador e equivalem aos picos observados na representação por curvas de nível. O pico próximo de um ponto projetado, ou máximo local, é encontrado seguindo o gradiente da função até o topo próximo. Sabemos que em uma combinação de Gaussianas, o gradiente da função composta será a soma vetorial dos gradientes parciais. Assim, podemos calcular o gradiente como na Equação 3.12:

$$\nabla F = \nabla G_1 + \nabla G_2 + \dots + \nabla G_n. \quad (3.12)$$

Os gradientes parciais precisam ser calculados separadamente. Para isso usamos as seguintes definições para a matriz de covariância ( $\Sigma$ ), a média ( $\mu$ ) e o ponto ( $X$ ) que se

deseja a probabilidade:

$$\Sigma = \begin{bmatrix} \sigma_1 & \sigma_2 \\ \sigma_3 & \sigma_4 \end{bmatrix}, X = \begin{bmatrix} x \\ y \end{bmatrix}, \mu = \begin{bmatrix} \mu_x \\ \mu_y \end{bmatrix}. \quad (3.13)$$

A função Gaussiana  $G(X)$  é definida como:

$$G(X) = \frac{1}{\sqrt{2\pi|\Sigma|}} e^{-\frac{1}{2}[X-\mu]^T \Sigma^{-1} [X-\mu]}, \quad (3.14)$$

onde encontramos o expoente  $B$  e o fator  $F$ :

$$B = -\frac{1}{2}[X - \mu]^T \Sigma^{-1} [X - \mu] \quad e \quad F = \frac{1}{\sqrt{2\pi|\Sigma|}}, \quad (3.15)$$

o que nos permite escrever a Equação 3.14 como:

$$G(X) = F e^B \quad (3.16)$$

e o seu gradiente como:

$$\nabla G(X) = \left[ F e^B \frac{dB}{dx}, F e^B \frac{dB}{dy} \right]. \quad (3.17)$$

O cálculo das derivadas  $\frac{dB}{dx}$  e  $\frac{dB}{dy}$  dependem da matriz inversa da covariância  $\Sigma^{-1}$ . Em nosso caso, temos uma matriz  $2 \times 2$  para a covariância e isso nos permite aplicar a matriz adjunta para conseguir a inversa rapidamente:

$$\Sigma^{-1} = \frac{1}{|\Sigma|} \cdot adj(\Sigma) \quad (3.18)$$

$$= \frac{1}{|\Sigma|} \cdot \begin{bmatrix} \sigma_4 & -\sigma_2 \\ -\sigma_3 & \sigma_1 \end{bmatrix}. \quad (3.19)$$

Substituindo a Equação 3.18 na Equação 3.15 temos:

$$B = -\frac{1}{2 \cdot |\Sigma|} (x^2 \sigma_4 + y^2 \sigma_1 - xy(\sigma_3 + \sigma_2)) \quad (3.20)$$

e, portanto, as derivadas nos respectivos eixos  $x$  e  $y$ :

$$\begin{aligned} \frac{dB}{dx} &= -\frac{1}{2 \cdot |\Sigma|} (2x\sigma_4 - y(\sigma_3 + \sigma_2)) \\ \frac{dB}{dy} &= -\frac{1}{2 \cdot |\Sigma|} (2y\sigma_1 - x(\sigma_3 + \sigma_2)). \end{aligned} \quad (3.21)$$

As posições dos jogadores são detectadas encontrando os máximos locais seguindo o gradiente. Cada ponto projetado no plano da quadra converge para o pico mais próximo. Em consequência das projeções das múltiplas câmeras, cada pico pode atrair múltiplos pontos gerando redundância na resposta do detector 3D. Para eliminar as redundâncias,

cada ponto convergido a um pico local é inserido em uma lista sem duplicatas. Ao final da análise de todos os pontos, o conteúdo da lista equivale ao conjunto das posições dos jogadores no plano da quadra.

Ao final, obtemos dois resultados importantes: uma função de probabilidades multimodal como resultado da fusão de múltiplas Gaussianas. As Gaussianas substituem as projeções dos pontos detectados nas imagens. O outro resultado é uma aproximação das posições dos jogadores pela subida pelo Gradiente.

## 3.2 Metodologia e Experimentos

O método de localização automática de jogadores de futebol de salão proposto possui uma fase dedicada ao treinamento que é de vital importância para os detectores, como discutido anteriormente. Também é necessário conhecer as funções que representam os erros de projeção das câmeras individualmente para modelar a função multimodal que descreve a probabilidade de um jogador ser encontrado em um ponto da superfície da quadra.

Dispomos de uma coleção de sete jogos gravados em FULL-HD usando quatro câmeras cada um, resultando em um total de quatorze vídeos considerando dois tempos em cada jogo. As câmeras foram dispostas ao redor da quadra de forma a cobrir toda a área do jogo com sobreposições que garantem a observação de um ponto da quadra por, pelo menos, duas das câmeras simultaneamente, tal como visto no Capítulo 1. Essa coleção corresponde a uma parte dos jogos do campeonato Sul-americano de Futebol de Salão Feminino do ano de 2009. Por questões de tempo no processamento, os vídeos em FULL-HD foram reduzidos para  $720 \times 480$  *pixels*.

Desse conjunto de vídeos, foi separado um tempo de um dos jogos para o treinamento. Um tempo de jogo é composto de quatro vídeos de aproximadamente 20 minutos, cada um gravado a 30 quadros por segundo, perfazendo um total de 36000 *frames* em cada câmera. Os demais conjuntos de vídeos foram utilizados para testes e validações.

Para a validação do método, é preciso um conjunto de dados anotados previamente, contendo indicações das posições reais dos jogadores para cada *frame* de um vídeo. O conjunto de anotações das posições reais dos jogadores foi obtido com a ajuda do sistema *Dvideo* desenvolvido pelo Laboratório de Biomecânica da Faculdade de Educação Física da UNICAMP. Esse é um sistema assistido que provê ferramentas para detecção e localização de jogadores de futebol em coordenadas no plano da quadra (Figuerola *et al.* [18]). O sistema vem sendo utilizado para anotar informações de posição e *scouting* [10]. Para a validação deste trabalho, foram anotados dois minutos de jogo em cada um dos 14 tempos disponíveis e foram considerados os 10 jogadores em quadra além dos dois árbitros.

O primeiro passo consiste em treinar os detectores. Esse treinamento foi realizado

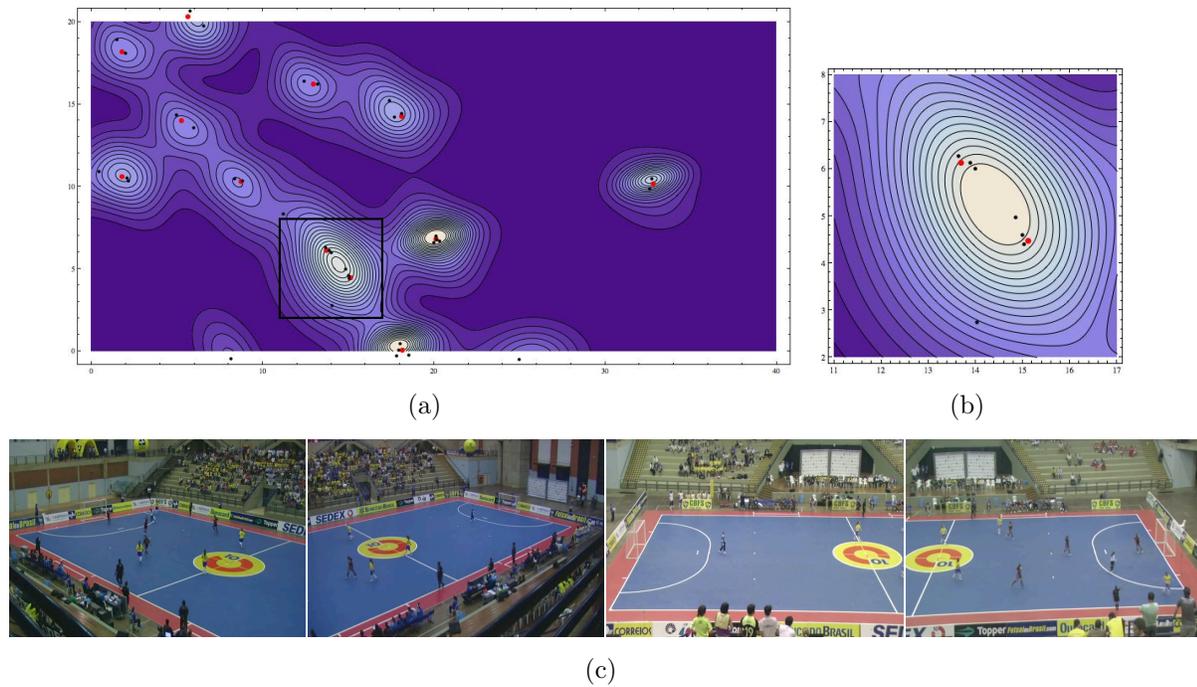


Figura 3.6: Curvas de nível ao redor dos pontos projetados. Em (a) os pontos vermelhos representam as marcações manuais e os pontos pretos representam pontos projetados das quatro câmeras sem distinção. Em (a) observamos as curvas de nível em toda a superfície da quadra, enquanto que (b) mostra a ampliação de uma região com jogadores próximos. Em (c) encontramos os quadros de origem nas quatro câmeras utilizadas.

usando uma coleção de amostras positivas e negativas colhidas do conjunto de treinamento. O conjunto de amostras positivas consiste de recortes de jogadores dentro da quadra vistos pelas quatro câmeras num total de 15.932 amostras. As amostras positivas foram obtidas manualmente com ajuda do software *imageclipper*. Este software foi desenvolvido com o objetivo de ajudar pesquisadores e desenvolvedores de visão computacional, a reunir rapidamente recortes de imagens para treinamento e testes. O conjunto de amostras negativas consiste de 18.400 recortes do plano de fundo em três diferentes tamanhos, ambos com aproximadamente o tamanho dos jogadores nas imagens. Estes recortes foram obtidos extraindo amostras das estimativas de plano de fundo. O plano de fundo foi estimado nas quatro câmeras usando a média entre os *pixels* dos *frames* encontrados nos vídeos de treinamento.

Após o treinamento dos detectores, é preciso computar os erros de projeção das câmeras medindo a média dos erros e as respectivas matrizes de covariância. O resultado das detecções nos vídeos de treinamento foram usados para o cálculo dos erros de projeção. A base média de cada retângulo foi projetada no plano da quadra gerando

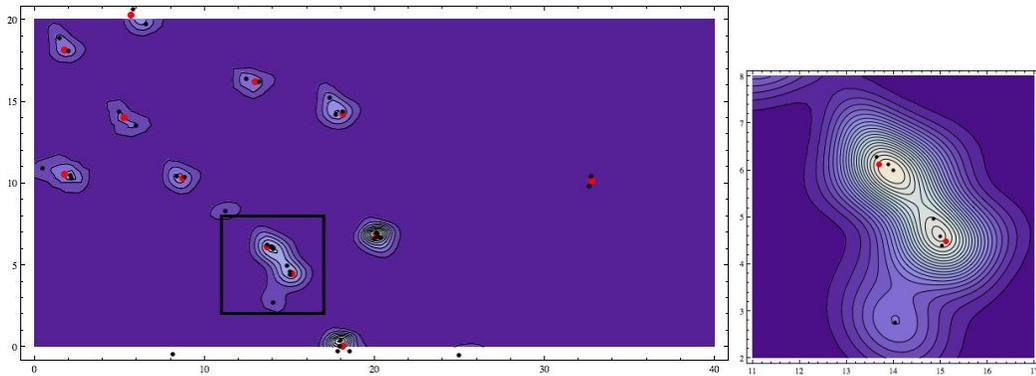


Figura 3.7: Curvas de nível obtidas por um fator de escala. A figura mostra os mesmos dados da Figura 3.6 com as covariâncias multiplicadas por um fator de 0,2.

uma coleção de pontos que deu origem a uma função multimodal, como apresentado na Seção 3.1.3.

A Figura 3.6 mostra um caso comum em jogos de futebol de salão em que existe proximidade entre os jogadores. Pelo fato de estarem próximos, as funções Gaussianas correspondentes também estarão próximas. Conseqüentemente, os picos das duas funções serão fundidos. Isso ocorre em função das matrizes de covariância que produzem Gaussianas de base larga e subida suave. Essas funções quando próximas se fundem de maneira a gerar um pico único. É possível reduzir esse efeito multiplicando as matrizes de covariância por um fator de escala menor que 1. Com isso, a base das Gaussianas é estreitada e os topos das Gaussianas são alongados. O resultado da fusão é uma função multimodal com picos bem definidos. A Figura 3.7 mostra o resultado obtido com um fator de escala igual a 0,2 escolhido empiricamente.

O método proposto busca posições em um plano que corresponde ao plano da quadra com base em imagens digitais. Por isso, a comparação com as anotações são baseadas em distância Euclidiana. O objetivo é mostrar que as posições estimadas ficam mais próximas das posições reais do que se usarmos apenas os detectores sem o uso da função multimodal proposta. As posições reais neste caso, são representadas pelas marcações manuais discutidas anteriormente.

Quando usamos apenas os detectores, projetamos em coordenadas do plano da quadra as bases médias dos retângulos encontrados nas imagens. Essas são as detecções no plano da quadra usando apenas os detectores. Em nosso caso, as detecções são representadas pelos picos encontrados na função multimodal usando a subida pelo gradiente. Cada uma das detecções, em ambos os casos, é relacionada à posição real mais próxima. As distâncias maiores que um metro são consideradas falso positivos (FP) e as menores são consideradas detecções corretas (DC). A diferença entre detecções corretas e o número de jogadores presentes na quadra equivale aos falsos negativos (FN). A Tabela 3.1 mostra a

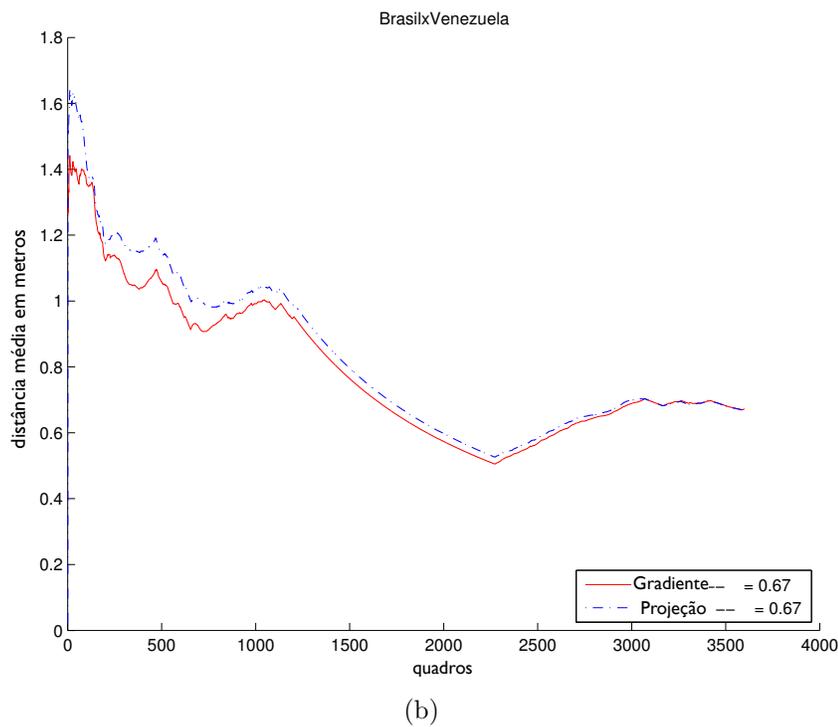
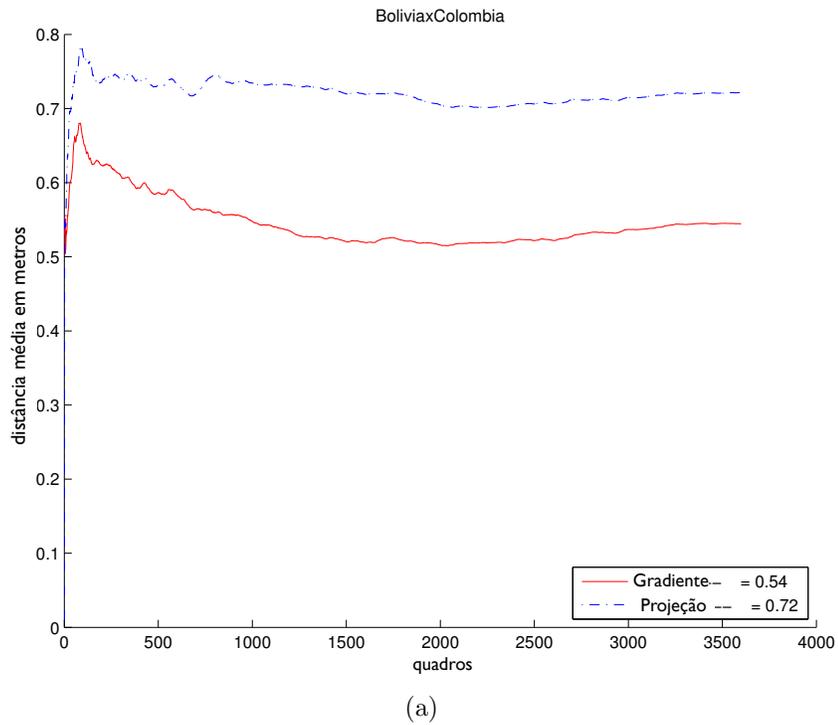


Figura 3.8: Erro médio na estimativa. As curvas mostram os erros médios acumulados por *frame* considerando apenas uma câmera. Em (a), mostramos o caso em que nossa técnica está mais próxima do real. Em (b), mostramos o caso em que tanto a estimativa usando apenas os detectores no plano da imagem e nossa técnica estão bem próximas.

Tabela 3.1: Contagem de DC, FP e FN para cada um dos vídeos usando a subida pelo gradiente e usando apenas os detectores.

Jogo	Tempo	Projeção			Gradiente		
		DC (%)	FP (%)	FN ( $\mu$ )	DC (%)	FP (%)	FN ( $\mu$ )
BoliviaxColombia	T-01	71.349	28.651	1.614	79.273	20.727	1.569
	T-02	69.976	30.024	2.502	81.537	18.463	2.426
BrasilxArgentina	T-01	77.063	22.937	2.242	83.009	16.991	2.353
	T-02	76.866	23.134	1.745	83.571	16.429	1.913
BrasilxColombia	T-01	56.727	43.273	3.442	62.664	37.336	3.622
	T-02	62.486	37.514	4.056	70.888	29.112	4.041
BrasilxPeru	T-01	72.406	27.594	1.861	79.224	20.776	1.981
	T-02	63.320	36.680	2.551	70.462	29.538	3.049
BrasilxVenezuela	T-01	70.443	29.557	1.092	70.670	29.330	2.012
	T-02	46.148	53.852	4.496	42.173	57.827	5.718
ColombiaxUruguai	T-01	67.198	32.802	1.107	73.171	26.829	1.473
	T-02	65.715	34.285	1.044	73.584	26.416	1.339
PeruxBolivia	T-01	57.479	42.521	5.957	61.351	38.649	5.881
	T-02	50.657	49.343	6.166	52.848	47.152	6.371

contagem de forma resumida. A coluna DC mostra a porcentagem das Detecções Corretas, a coluna FP mostra a porcentagem de Falsos Positivos e a coluna FN motra a média de Falsos Negativos por quadro de vídeo.

A distância de uma posição real a uma detecção considerada correta mostra o erro da estimativa medido em metros. A Figura 3.8 mostra o melhor e o pior caso de nossas comparações baseadas em distância. Em ambos os casos, exibimos os erros médios acumulados em cada *frame*. Optamos pelo cálculo separado por câmera, já que os erros de projeção são diferentes para cada câmera o que iria alterar o resultado. O Apêndice A mostra uma coleção com todos os gráficos da relativos à Tabela 3.2.

A Tabela 3.2 mostra os erros médios acumulados finais e seus respectivos desvios. Na grande maioria dos casos, nossa técnica obteve melhoria na estimativa das posições quando comparada com o uso apenas dos detectores no plano da imagem. Além disso, o uso dos detectores sem uma técnica para fundir as projeções leva a detecções múltiplas do mesmo jogador, bem como, dificulta o uso da técnica em um sistema de rastreamento por exemplo, ou ainda em tomadas de decisão por parte da equipe técnica onde a estimativa de posição precisa ser única.

Tabela 3.2: Erros médios descritos em metros e desvio encontrados nas respectivas câmeras.

Jogo	Tempo	Câmera	Gradiente		Projeção	
			$\mu$	$\sigma$	$\mu$	$\sigma$
Bolívia x Colômbia	T-01	1	0.58	0.13	0.73	0.16
		2	0.68	0.19	0.78	0.20
		3	0.75	0.24	0.88	0.21
		4	0.53	0.14	0.65	0.15
	T-02	1	0.65	0.21	0.77	0.21
		2	0.76	0.18	0.91	0.19
		3	0.54	0.13	0.72	0.13
		4	0.76	0.20	0.91	0.20
Brasil x Argentina	T-01	1	0.71	0.21	0.83	0.20
		2	0.74	0.27	0.79	0.27
		3	0.72	0.24	0.81	0.24
		4	0.55	0.16	0.63	0.15
	T-02	1	0.84	0.27	0.91	0.25
		2	0.63	0.19	0.68	0.19
		3	0.82	0.24	0.90	0.23
		4	0.79	0.25	0.89	0.23
Brasil x Colômbia	T-01	1	0.78	0.21	0.82	0.20
		2	0.81	0.23	0.90	0.23
		3	0.78	0.25	0.90	0.22
		4	0.80	0.21	0.84	0.21
	T-02	1	0.81	0.22	0.87	0.19
		2	0.77	0.21	0.81	0.19
		3	0.88	0.27	1.00	0.26
		4	0.67	0.19	0.79	0.19
Brasil x Peru	T-01	1	1.10	0.35	1.12	0.32
		2	0.66	0.19	0.73	0.20
		3	0.64	0.20	0.74	0.19
		4	1.06	0.36	1.10	0.30
	T-02	1	0.97	0.25	1.03	0.22
		2	0.89	0.23	0.93	0.22
		3	0.98	0.23	1.01	0.21
		4	0.91	0.22	0.96	0.19

Tabela 3.2: Erros médios descritos em metros e desvio encontrados nas respectivas câmeras.

Jogo	Tempo	Câmera	Gradiente		Projeção	
			$\mu$	$\sigma$	$\mu$	$\sigma$
Brasil x Venezuela	T-01	1	0.50	0.15	0.66	0.17
		2	0.58	0.25	0.82	0.24
		3	0.36	0.18	0.53	0.17
		4	0.45	0.14	0.52	0.14
	T-02	1	0.81	0.64	0.85	0.65
		2	0.95	0.59	0.89	0.61
		3	0.79	0.61	0.83	0.64
		4	0.67	0.67	0.67	0.68
Colombia x Uruguai	T-01	1	0.90	0.27	1.04	0.25
		2	0.77	0.24	0.93	0.23
		3	0.56	0.16	0.65	0.16
		4	0.74	0.25	0.71	0.23
	T-02	1	0.73	0.16	0.89	0.17
		2	0.78	0.19	0.86	0.21
		3	0.86	0.24	0.93	0.23
		4	0.63	0.16	0.62	0.13
Peru x Bolívia	T-01	1	0.68	0.25	0.77	0.25
		2	0.79	0.25	0.81	0.25
		3	0.94	0.30	1.03	0.28
		4	0.77	0.35	1.03	0.30
	T-02	1	0.71	0.23	0.80	0.24
		2	0.84	0.26	0.80	0.24
		3	0.93	0.29	1.00	0.32
		4	0.80	0.28	0.86	0.26



## Capítulo 4

# Rastreamento de Jogadores no Plano da Quadra

No Capítulo 3, apresentamos um algoritmo para encontrar uma estimativa das posições dos jogadores no plano do jogo a cada instante. Neste capítulo, usaremos parte do resultado obtido anteriormente como passo de observação para confirmar a predição de um filtro preditivo responsável pelo rastreamento.

O Capítulo 3 descreveu o uso de detectores treinados para encontrar jogadores nas imagens e localizá-los na quadra, usando homografia. Uma representação 2D dos jogadores é projetada em um plano virtual representando a quadra e os dados multicâmera são fundidos em uma função bidimensional e multimodal usada para encontrar um jogador em uma região da quadra. Para tratar possíveis erros de projeção, o método descreve as posições dos jogadores no plano da quadra utilizando uma representação probabilística. Portanto, a função multimodal representa a probabilidade de um jogador ser encontrado em uma posição  $S(x, y)$  da quadra, composta por uma mistura de Gaussianas de parâmetros conhecidos.

O método de rastreamento proposto irá incorporar a função multimodal como modelo de observação para ponderar as partículas do filtro de partículas e, assim, computar a trajetória em coordenadas do plano da quadra. O resultado é um método para rastreamento dos jogadores de futebol de salão em coordenadas do plano da quadra durante uma partida, com base apenas em vídeos adquiridos por múltiplas câmeras. Para isso, estamos considerando que as câmeras são estacionárias e também conhecemos um conjunto de pontos característicos do plano de jogo, possibilitando o uso da Homografia.

O restante desse capítulo está organizado em quatro seções. A Seção 4.1 formaliza conceitos importantes para o desenvolvimento do restante do texto. A Seção 4.2 introduz os passos do método de rastreamento proposto. A Seção 4.3 apresenta a metodologia e os experimentos.

## 4.1 Conceitos Básicos

O problema de rastreamento visual de objetos consiste em processar uma sequência de imagens para descrever o movimento de um ou mais objetos em uma cena [22] ou, ainda, gerar inferência sobre o movimento de um objeto a partir de uma sequência de imagens [19]. Além de recuperar as posições de um objeto em um intervalo de tempo, é possível rastrear informações que precisam de medição indireta, como temperatura de alto forno, que pode ser medida por análise de cor, ou ainda, um sistema de rastreamento poderia ser usado para mostrar a evolução da temperatura de um forno durante certo período de tempo, dentre outras aplicações.

Normalmente, o rastreamento se refere ao histórico de posições de um objeto observado, além de outras informações que podem ser obtidas utilizando informação visual, como velocidade instantânea, aceleração, temperatura, e qualquer outra medida que possa ser deduzida a partir de informações encontradas nas imagens.

Para rastrear um objeto, é preciso detectá-lo na imagem. Alguns métodos processam a imagem inteira para localizar o objeto rastreado [18, 30]. A cada quadro de uma sequência de vídeo, todos os *blobs* existentes são detectados para então se decidir qual *blob* está sendo monitorado. Técnicas mais recentes utilizam filtros preditivos capazes de usar um modelo de movimento chamado de dinâmica e, com isso, estimar o estado ou a posição de um objeto no próximo quadro da sequência. Essa estimativa é chamada de predição. A predição é ajustada por uma observação local que não precisa levar em conta toda a imagem de entrada. O ajuste é necessário uma vez que não se sabe exatamente como o objeto se move em um dado instante. Por isso, os erros se acumulam e podem levar a predições futuras que não representam o objeto de interesse. Enquanto o objeto é observável, a observação é usada para ajustar parâmetros da predição para manter boas estimativas. Quando o objeto não é observável, por exemplo, em um momento de oclusão, o rastreador usa apenas os dados da predição para manter o rastreamento até que o objeto seja detectado novamente. Maiores detalhes a respeito de filtros preditivos podem ser encontrados em [23]. Dois filtros amplamente utilizados na literatura são os filtros de Kalman [1, 23, 62] e os filtros de partículas [23, 26].

Os filtros de Kalman usam funções Gaussianas para estimar o próximo estado em um rastreamento. A vantagem é que a função Gaussiana é uma função paramétrica de fácil representação [23, 59]. Tais filtros possuem a vantagem de conseguir estimar estados passados, presentes e futuros em uma sequência ou em uma função, o que lhe dá o poder de inferir estados perdidos ou que não puderam ser detectados. Isso ocorre, por exemplo, quando o objeto rastreado é sobreposto por outro elemento da cena causando uma oclusão. O problema nessa abordagem é que a função Gaussiana é unimodal e, conseqüentemente, o filtro não consegue representar múltiplas hipóteses simultaneamente.

Por outro lado, os filtros de partículas como aqueles apresentados por Isard e Blake [26], representam suas funções de probabilidades de forma não paramétrica, usando conjuntos de amostras ponderadas chamadas *partículas*. Esse tipo de representação permite que funções desconhecidas e multimodais sejam representadas. No conjunto, cada partícula  $X_t$  é associada a um peso  $W_t$  que representa o valor da função no ponto determinado pela partícula naquele instante de tempo. Quanto maior o número de partículas, melhor a representação da função desejada. Esse modelo de funções não paramétricas é usado para representar a função de probabilidades do objeto ocorrer no próximo quadro de uma sequência, antes que o quadro seja observado. Essa função é chamada de função de probabilidade *a priori* e é usada para estimar o estado do objeto no próximo quadro.

Em um filtro de partículas, um estado é a representação do objeto rastreado em cada instante de tempo ou em cada quadro de uma sequência observada. Um estado pode conter qualquer informação considerada relevante ou que se deseja rastrear. Normalmente, representamos os estados na forma vetorial. Por exemplo, se a tarefa é rastrear a posição, velocidade e temperatura de um objeto, cada estado pode ser composto por um vetor de três valores que correspondem a tais informações.

A função de probabilidade *a priori* é uma estimativa do que deve ocorrer no próximo quadro com base em uma dinâmica de movimento conhecida. Quando o filtro tem acesso ao novo quadro, é preciso ajustar ou confirmar a função *a priori* que representa as estimativas. Nesse ponto, cada partícula passa por uma fase de *observação* que verifica sua representatividade naquele quadro ajustando o peso  $W_t$  de cada partícula. A nova representação ponderada da função corresponde à probabilidade de uma partícula  $X_t$  ocorrer, dada uma observação  $Z_t$ , sendo denotada por  $P(X_t|Z_t)$ . Essa nova função é chamada de função de probabilidade *a posteriori*, pois foi confirmada pela observação.

O rastreamento é um processo contínuo e portanto, sempre necessita de uma estimativa para o quadro seguinte. A estimativa para o próximo passo é construída sorteando partículas do conjunto atual, permitindo repetição e dando preferência às partículas de maior peso. O novo conjunto de partículas precisa possuir o mesmo tamanho do anterior. Os valores das novas partículas são ajustados considerando uma dinâmica de movimento conhecida para criar uma representação do que será encontrado no próximo passo. Entretanto, existem repetições que podem levar o sistema a um colapso. Assim, um erro aleatório que representa o nível de incerteza no processo é adicionado a cada partícula. A adição do erro aleatório separa as partículas repetidas e melhora a representatividade da função naquele ponto. Esse passo é chamado de *predição*, pois, com base no conhecimento do movimento do objeto rastreado, estima-se o próximo conjunto de partículas acrescentando um erro aleatório para garantir a distinção entre as amostras. A nova função representada pelo novo conjunto de partículas é a função *a priori*  $P(X_{t+1}|Z_t)$  para o próximo quadro da sequência.

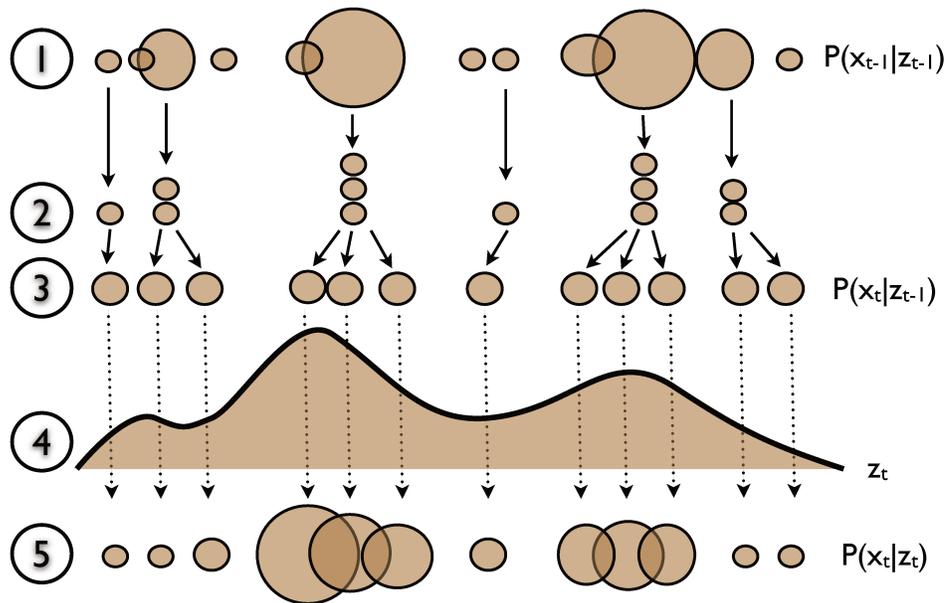


Figura 4.1: Passos do filtro de partículas. Cada camada mostra uma das etapas: (1) função *a posteriori* em  $t - 1$ ; (2) amostragem; (3) propagação, resultando na função *a priori*  $P(X_t|Z_{t-1})$ ; (4) observação em  $t$ ; e (5) função *a posteriori*  $P(X_t|Z_t)$  em  $t$ .

A Figura 4.1 mostra os passos de um ciclo do filtro de partículas, iniciando na função *a posteriori* no tempo  $t - 1$  e terminando na função *a posteriori* no tempo  $t$ . A primeira camada mostra um conjunto de partículas representadas por elipses de tamanhos diferentes. No desenho, o tamanho da elipse representa o peso  $W$  da partícula no processo. A segunda camada mostra um conjunto de partículas com algumas repetições, sendo o resultado da etapa de Amostragem. Nesse ponto, os pesos das partículas perderam seu significado e precisam ser medidos novamente, por isso, as elipses têm o mesmo tamanho. A terceira camada da figura mostra o resultado da propagação, onde as partículas sorteadas são submetidas à dinâmica acrescida de um erro aleatório. Nesse ponto, o conjunto de partículas representa a estimativa para o próximo passo da sequência. Na quarta camada, a medição está representada como uma função em que a altura representa o peso da medição naquele ponto. As partículas encontradas em regiões de maior peso na função de observação terão pesos maiores e serão representadas por elipses maiores. Após o passo de medição, o conjunto de partículas da quinta camada representa uma função de probabilidades *a posteriori* com as partículas ponderadas no tempo  $t$  e está pronta para um novo ciclo. Esse processo é repetido até o final do vídeo observado.

Neste capítulo, apresentamos uma abordagem para o problema de rastreamento de jogadores de futebol de salão baseada nos filtros de partículas. A Seção 4.2 apresenta mais detalhes do método.

## 4.2 Método proposto

Neste capítulo, propomos um método de rastreamento de jogadores de futebol de salão baseado em um filtro de partículas com um modelo de observação probabilístico e robusto. A nossa contribuição está no rastreamento diretamente em coordenadas do mundo real, usando um modelo de observação que combina dados detectados em múltiplas câmeras estacionárias. Assim como mostra a Figura 4.1, partimos de um conjunto de partículas inicial e, a cada instante, atualizamos o conjunto seguindo as etapas de reamostragem, propagação e observação, detalhados a seguir.

### 4.2.1 Reamostragem

Como mostrado anteriormente, um filtro de partículas é um processo iterativo que refaz um conjunto de amostras ponderadas a cada ciclo. A esse processo dá-se o nome de reamostragem (*resampling*) porque refaz um conjunto de amostras com base no conjunto existente. A cada novo conjunto, tem-se uma nova representação da função modelada. As iterações do filtro mostram a evolução da função ao longo do tempo. O objetivo é replicar partículas com maiores pesos (mais importantes) e também descartar aquelas que possuem pouca probabilidade de ocorrer.

As iterações do filtro iniciam com uma representação arbitrária da função, ou seja, o filtro inicia com um conjunto arbitrário de partículas dado por uma inicialização do sistema. É comum o uso de pesos iguais para as partículas no conjunto de inicialização. Isso significa que elas são equiprováveis [26].

A reamostragem é o passo na iteração do filtro que escolhe as partículas do conjunto no tempo  $t - 1$  para construir o conjunto no tempo  $t$ . A escolha é feita por um sorteio que permite repetições e favorece as partículas de maior peso.

A implementação consiste em mapear as amostras em um intervalo  $[0, 1]$  e atribuir porções maiores do intervalo a partículas de maior peso. Finalmente, sorteia-se um número aleatório  $\eta$  de acordo com uma distribuição uniforme. A partícula correspondente ao intervalo que contém o  $\eta$  sorteado será escolhida. Esse efeito pode ser obtido usando a função integral.

Quando fazemos a integral de uma função, estamos calculando a área ocupada pela região entre o eixo  $x$  e a curva da função, dado um intervalo fechado. Em uma integral discreta, calculamos a soma dos valores discretos de uma função dentro do intervalo. Se considerarmos uma Gaussiana como a função amostrada, as partículas que precisam de maior preferência no sorteio serão aquelas mais próximas da média.

A Figura 4.2 mostra o gráfico de uma Gaussiana de média zero e desvio padrão igual a três na Figura 4.2(a) e sua integral na Figura 4.2(b). Podemos dizer que os maiores valores em 4.2(a) estão no intervalo  $[-3, 3]$  do eixo  $x$ , menos de  $1/3$  do intervalo total

$[-10, 10]$ . Na Figura 4.2(b), observamos que o mesmo intervalo no eixo  $x$  produz uma imagem no intervalo  $[0.201, 0.8792]$ , aproximadamente 70% do intervalo total  $[0, 1]$ . Isso acontece porque os valores correspondente em 4.2(a) são os que causam maior crescimento na função da Figura 4.2(b).

Mapeando as partículas correspondentes ao eixo  $x$  na Figura 4.2(a) aos respectivos valores no eixo  $y$  da Figura 4.2(b) obtemos um mapeamento das partículas em um intervalo  $[0, 1]$  onde o valor  $y$  mapeado indica o tamanho da porção que deveria ocupar.

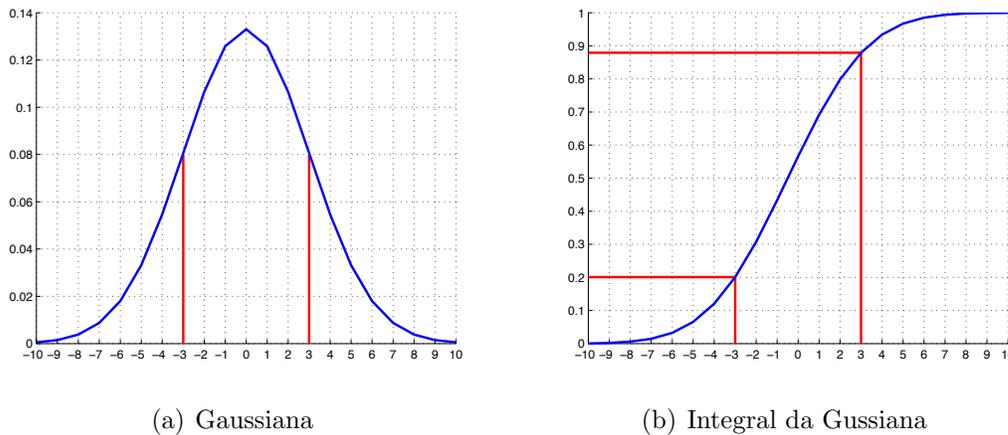


Figura 4.2: Exemplo de reamostragem. A figura mostra um exemplo de função Gaussiana (a) e sua integral (b). Na reamostragem do filtro de partículas, o sorteio é feito no espaço das imagens na função integral e aproximado para a ocorrência mais próxima. A partícula sorteada é aquela que gerou o valor correspondente ao sorteio. Pela figura, é fácil perceber que as partículas de maior peso são privilegiadas pois estão espalhadas no espaço das imagens da função integral.

Podemos usar esta informação para sortear as partículas do novo conjunto favorecendo as de maior peso. Se escolhermos um número aleatório  $\eta$  no intervalo  $[0, 1]$  correspondendo ao eixo  $y$  da Figura 4.2(b), teremos mais chance de escolher um  $y$  na região mais espalhada do que escolher um  $y$  na região mais densa.

O sorteio de  $N$  partículas usa um vetor de tamanho  $N$  cujos valores são valores acumulados dos pesos das partículas, assim como ocorre em uma integral discreta. Um número aleatório é escolhido no intervalo  $[0, 1]$  e uma busca binária determina qual a posição do vetor foi escolhida. A partícula cujo peso gerou o valor escolhido será a partícula sorteada. Esse mecanismo de escolha privilegia os maiores pesos mas não impede que uma partícula de peso baixo seja escolhida.

Após o passo de reamostragem, um novo conjunto de partículas é criado para representar a função no próximo ciclo do filtro. Cada ciclo representa um instante de tempo

no processo de rastreamento e precisa considerar o movimento do objeto rastreado. A próxima seção mostra como o movimento do objeto influencia no conjunto das partículas.

### 4.2.2 Propagação

Como discutido na Seção 4.1, cada partícula de um conjunto é um possível estado que o objeto pode assumir. Em nosso caso, o objeto rastreado é um jogador durante o jogo e seu estado consiste de sua posição  $S(x, y)$  no plano da quadra. Sabemos que nosso objeto não é estático na cena, por isso, é preciso representar uma dinâmica de movimento que descreve como o objeto se move.

Para tratar esse problema, adotamos uma dinâmica simples de movimento uniforme como mostram as Equações 4.1 e 4.2, já que a variação entre os *frames* é muito pequena. Nesta dinâmica, a variação de tempo equivale à frequência de gravação dos vídeos analisados (30 quadros por segundo). O novo estado  $S_t$  é uma estimativa de posição considerando a variação de tempo entre os quadros do filme:

$$S_t = S_{t-1} + V_{t-1}\Delta(t) \quad (4.1)$$

$$S_t = \begin{bmatrix} 1 & 0 & 1/30 & 0 \\ 0 & 1 & 0 & 1/30 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} x \\ y \\ v_x \\ v_y \end{bmatrix}_{t-1}. \quad (4.2)$$

Para estimar a nova posição  $S_t$ , é preciso saber a velocidade em que o objeto está se movendo em  $t - 1$ . Por isso, apesar de estarmos interessados na posição  $(x, y)$  do objeto, o estado  $S$  mantém uma estimativa de velocidade  $(v_x, v_y)$  para garantir o movimento das partículas. Como pode ser observado nas Equações 4.1 e 4.2, a velocidade em  $t - 1$  é mantida em  $t$  e as partículas escolhidas com repetição no passo de reamostragem (4.2.1) continuam idênticas.

Nesse ponto do processo, acrescentamos um erro aleatório ao sistema para que as partículas idênticas em  $t - 1$  se tornem diferentes em  $t$ . Em nosso caso, simplesmente acrescentamos um erro à velocidade das partículas, dando variação ao movimento das respectivas partículas e tornando-as diferentes para evitar um colapso do sistema. A Equação 4.3 mostra a propagação do sistema com o acréscimo do erro:

$$S_t = \begin{bmatrix} 1 & 0 & 1/30 & 0 \\ 0 & 1 & 0 & 1/30 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} * \left( \begin{bmatrix} x \\ y \\ v_x \\ v_y \end{bmatrix}_{t-1} + \begin{bmatrix} 0 \\ 0 \\ e_x \\ e_y \end{bmatrix} \right). \quad (4.3)$$

Além da dinâmica do sistema, precisamos ponderar as partículas e construir a função *a priori*. Para isso, utilizamos uma função multimodal e bidimensional que nos dá a probabilidade de encontrar um jogador em uma dada posição da quadra. A Seção 4.2.3 descreve tal função obtida no Capítulo 3.

### 4.2.3 Função de Observação

No Capítulo 3, propomos um método para localizar os jogadores no plano da quadra. O método cria uma função multimodal e bidirecional que retorna a probabilidade de um jogador ser encontrado em uma região da quadra. A função multimodal é usada em nosso trabalho como modelo de observação do Filtro de Partículas e pode ser resumida em três estágios principais a seguir, tal como mostrado na Figura 4.3.

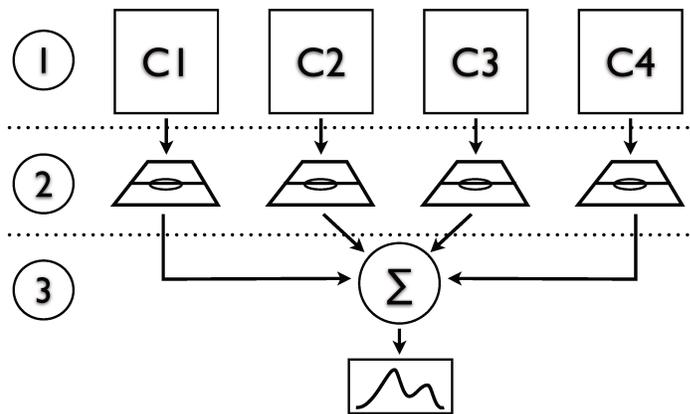


Figura 4.3: Estágios principais do modelo de observação. Em 1, os jogadores são detectados no plano da imagem usando, em cada uma das câmeras, um detector treinado para encontrar jogadores de futebol de salão no plano da imagem. Em 2, as observações feitas são projetadas no plano da quadra usando homografia. Em 3, os dados projetados são combinados em uma função de probabilidade bidimensional e multimodal que retorna a probabilidade de ser encontrado um jogador em uma dada posição da quadra.

#### Estágio 1 - Detecção dos Jogadores no Plano das Imagens

Esse estágio utiliza um detector de objetos treinado para encontrar jogadores de futebol de salão no plano das imagens independentemente em cada câmera. Alguns métodos de detecção de objetos podem ser usados como, por exemplo, o método de Viola e Jones [61] e o método de Felzenszwalb et al. [14].

Neste trabalho, usamos o modelo de Viola e Jones [61] treinado para a detecção de jogadores de futebol de salão. Tal detector é baseado em características do tipo Haar cujo valor é calculado pela diferença entre as somas dos *pixels* de retângulos adjacentes.

O método utiliza janelas deslizantes para detectar objetos. As características do tipo Haar são vistas como classificadores fracos. Durante o treinamento, os detectores fracos são organizados em cascata. A cascata de classificadores é organizada de forma a eliminar rapidamente, nos primeiros estágios, as janelas que não representam um objeto de interesse, deixando para as etapas seguintes e de maior custo computacional, apenas aquelas com maior potencial de representar um objeto procurado. O cálculo das características é acelerado usando o conceito de imagem integral *ii*. Maiores detalhes do método podem ser encontrados em [61].

### **Estágio 2 - Projeção das Detecções no Plano da Quadra**

Os objetos de interesse se movem sobre o plano da quadra do jogo e estão situados sobre um plano no mundo 3D. Podemos usar a homografia de pontos específicos (e.g., pés dos jogadores) para encontrar suas localizações no plano da quadra. Cada jogador encontrado pelo detector é representado por um retângulo no plano da imagem. Em nosso trabalho, consideramos a base média desse retângulo como uma boa representação da localização do pé do jogador no plano da imagem.

Para fazer a homografia, podemos utilizar pontos característicos da quadra cujas coordenadas conhecemos a priori. Esses pontos conhecidos, como a marca de pênalti e as quinas da quadra entre outros, servem para encontrar uma função de mapeamento entre os pontos no plano da imagem e os pontos no plano da quadra.

### **Estágio 3 - Fusão dos dados multi-câmera**

Após a projeção dos pontos detectados nas imagens no plano da quadra, temos um conjunto global de detecções descrito em coordenadas do mundo. Devido a possíveis erros acumulados tanto pela detecção quanto pela projeção, esses pontos representam uma estimativa da localização de cada jogador e não sua posição real.

Considerando os pontos projetados como regiões com potencial de corresponder à localização de um jogador, podemos substituir cada ponto por uma função Gaussiana 2D com parâmetros conhecidos. Em nosso trabalho, representamos o plano da quadra com uma função multimodal única, composta por uma mistura de Gaussianas que correspondem aos pontos projetados pelas múltiplas câmeras. Cada Gaussiana representa a incerteza na projeção do ponto correspondente e possui média (erro médio da projeção) e covariância que variam de acordo com cada câmera.

Podemos calcular os parâmetros das Gaussianas a partir de um pequeno conjunto de vídeos de treinamento que possuem as posições dos jogadores manualmente anotadas em coordenadas do plano da quadra. Sabendo as posições reais de cada jogador nos vídeos de treinamento, podemos comparar cada detecção com seu correspondente no conjunto

anotado. A correspondência é dada pela menor distância.

A partir das correspondências dos pontos detectados e anotados no treinamento, podemos calcular o erro médio de projeção nas direções  $x$  e  $y$  e covariância do erro em cada câmera. Desse modo, substituímos cada ponto projetado por uma Gaussiana com média igual à sua posição ajustada pelo erro médio e covariância calculados para a câmera que gerou o ponto.

Finalmente, podemos representar o conjunto global de pontos projetados por uma função multimodal, composta por uma mistura de Gaussianas cujos parâmetros são calculados da forma descrita acima. A função é uma representação do plano da quadra em que podemos medir a probabilidade de encontrar um jogador em um dado ponto e pode ser diretamente utilizada na etapa de observação do filtro de partículas fechando o modelo. A Figura 4.4 mostra uma representação gráfica de uma das observações do sistema, para um jogo analisado.

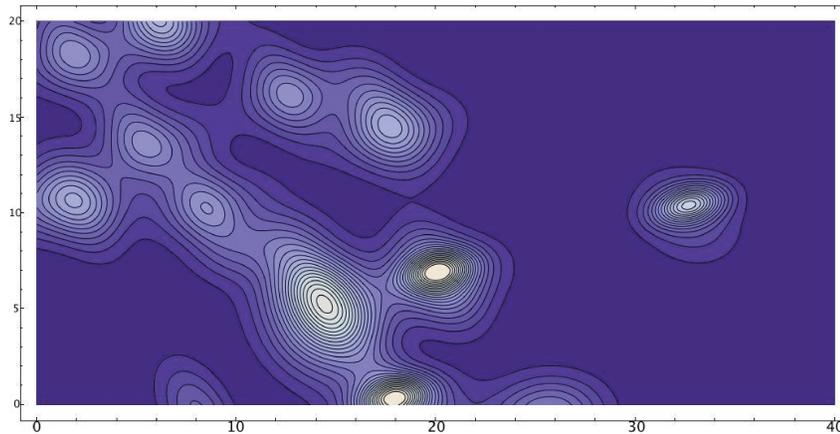


Figura 4.4: Observação do sistema de rastreamento em um dos ciclos do filtro de partículas. As projeções multicâmera são substituídas por funções Gaussianas com parâmetros previamente treinados. Como resultado, temos uma função 2D que retorna a probabilidade de se encontrar um jogador em uma determinada região.

### 4.3 Metodologia e experimentos

Para validar nosso método, utilizamos vídeos de jogos de futebol de salão gravados por um conjunto de quatro câmeras estacionárias posicionadas ao redor da quadra como descrito no Capítulo 1.

Dispomos de uma coleção de sete jogos gravados em FULL-HD usando quatro câmeras cada um a 30 quadros por segundo. Cada jogo possui dois tempos. Para simplificação, consideramos cada tempo como um jogo e portanto 14 jogos gravados por quatro câmeras.

Essa coleção foi gravada durante o *Campeonato Sul-americano de Futebol de Salão Feminino* em 2009.

Para os testes, consideramos apenas os dois primeiros minutos de cada jogo. Esta limitação é devida à dificuldade de anotar, manualmente, as posições dos jogadores em coordenadas do plano da quadra para posterior análise da qualidade do rastreamento.

Para o correto funcionamento do filtro de partículas, precisamos iniciar o conjunto de partículas a partir das posições iniciais dos jogadores que queremos rastrear. Para isso, precisamos marcar a posição dos jogadores de interesse no primeiro quadro de cada vídeo manualmente. Em nosso caso, estamos interessados nos 10 jogadores em quadra e também nos dois árbitros. Para cada objeto de interesse, iniciamos um filtro com um conjunto de 500 partículas cada um.

Em nossa representação, um rastreador é apresentado como um conjunto de pontos azuis que correspondem às partículas de seu conjunto e um número que corresponde a seu Identificador no sistema. A Figura 4.5 mostra o mapa da quadra com as representações de cada rastreador.

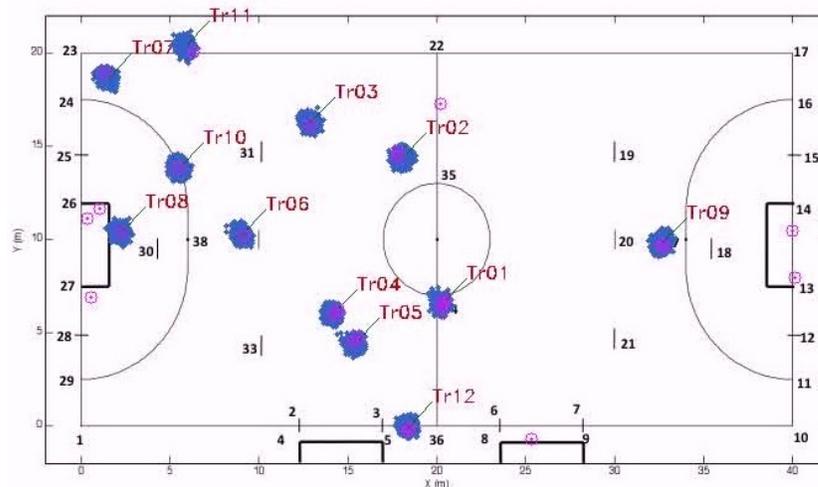
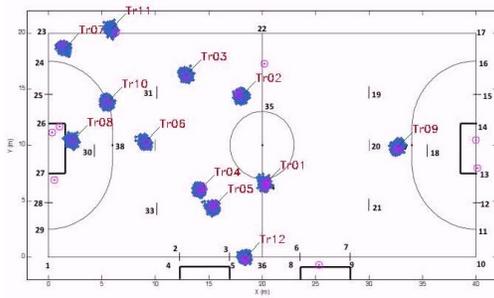
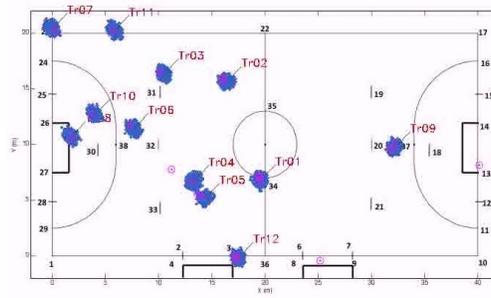


Figura 4.5: Inicialização dos Rastreadores. Os pontos azuis representam as partículas de cada rastreador. Os rastreadores iniciados são representados por seu número de identificação. Podemos observar também, algumas marcações que correspondem a detecções equivocadas que não interferem no rastreamento.

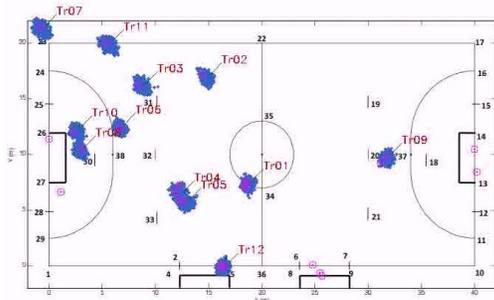
A cada novo quadro, o rastreamento executa um novo ciclo e refaz seu conjunto de partículas para determinar a nova posição do objeto rastreado. Como posição do objeto rastreado usamos a média ponderada das partículas. Na Figura 4.6, apresentamos uma sequência de alguns quadros ilustrando o comportamento do rastreador. É possível perceber que o rastreamento se mantém ao longo do tempo.



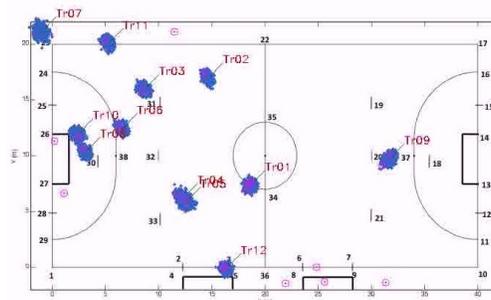
(a) quadro 007



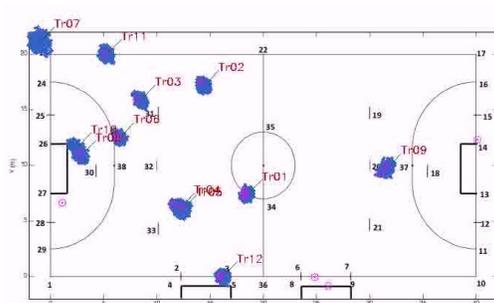
(b) quadro 030



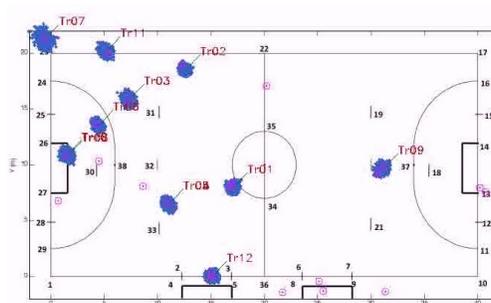
(c) quadro 049



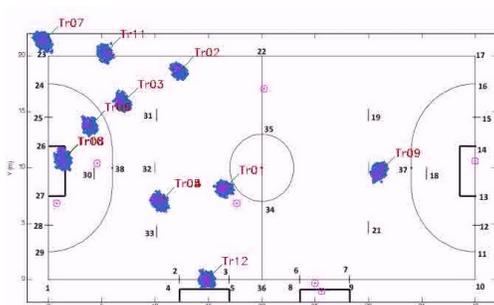
(d) quadro 052



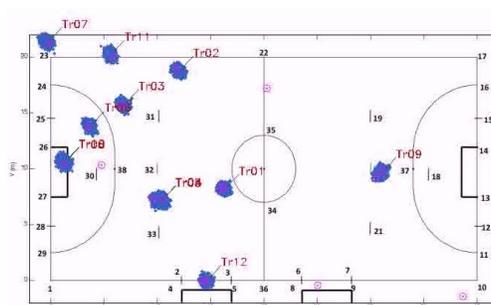
(e) quadro 055



(f) quadro 078

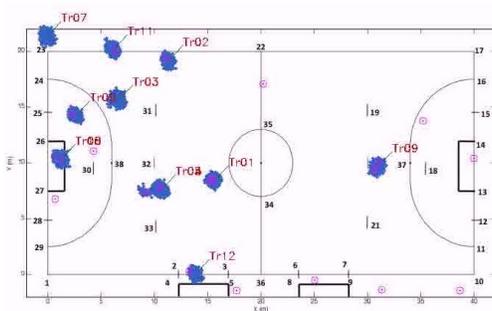


(g) quadro 085

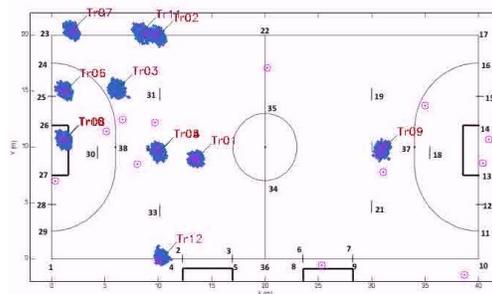


(h) quadro 089

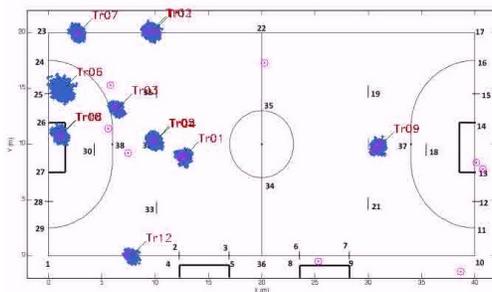
Figura 4.6: Demonstração do rastreamento. Na Figura é possível observar que o rastreamento é mantido ao longo dos quadros do vídeo. Os quadros foram escolhidos de maneira a ilustrar tanto o rastreamento correto quanto os momentos de confusão no rastreamento.



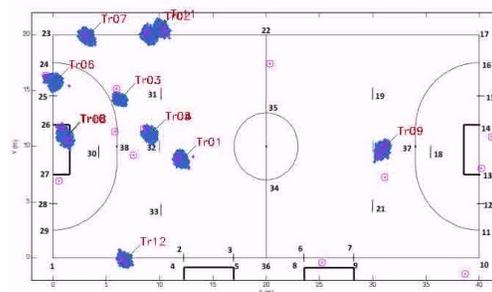
(a) quadro 100



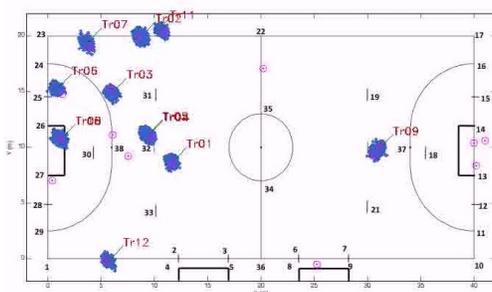
(b) quadro 134



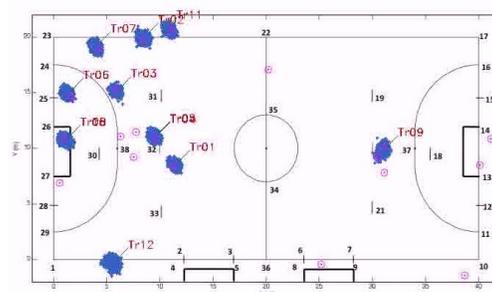
(c) quadro 151



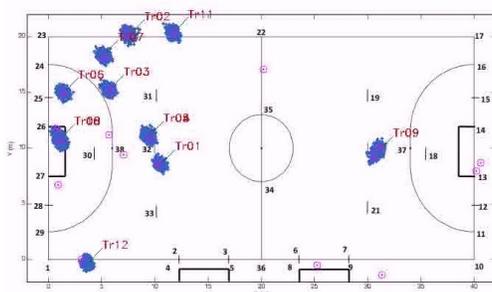
(d) quadro 161



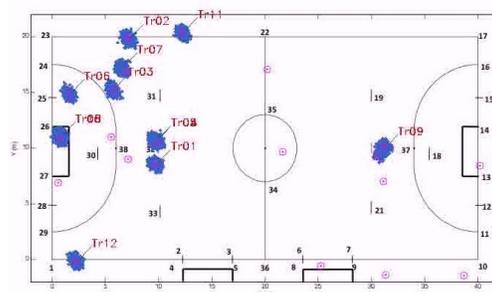
(e) quadro 169



(f) quadro 175



(g) quadro 190



(h) quadro 209

Continuação da figura 4.6.

Durante os testes, obtivemos trajetórias encontradas por rastreamentos bem sucedidos e outras resultantes de confusões entre jogadores que se aproximaram durante o rastreamento.

Nas situações em que as trajetórias de dois ou mais jogadores se cruzam, as confusões causadas pela proximidade das trajetórias são resolvidas pela predição do filtro baseada na dinâmica dos objetos. No entanto, quando as trajetórias de dois ou mais jogadores tendem a ser coincidentes por alguns instantes, temos uma situação de confusão.

Uma situação de confusão pode ocorrer, por exemplo, em uma disputa de bola ou em uma comemoração. Nesses casos, os jogadores ficam muito próximos uns dos outros e se movem na mesma direção. A Figura 4.6 mostra um exemplo de confusão em que os jogadores observados estão se movendo na mesma direção e por isso confundem o filtro. Essa configuração faz com que o filtro dependa da observação para se ajustar e conseguir distinguir as múltiplas hipóteses.

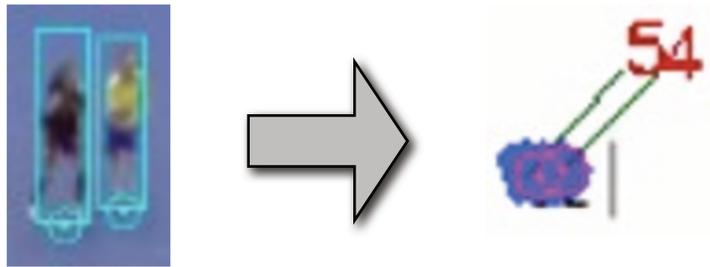
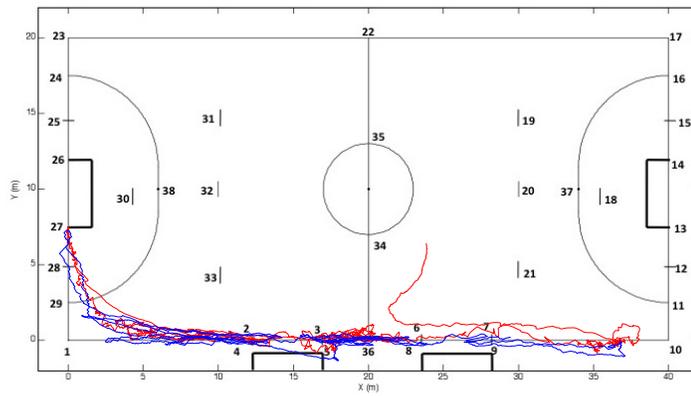


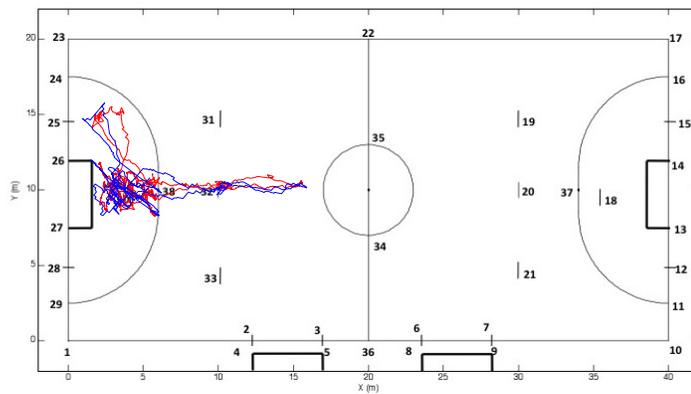
Figura 4.6: Exemplo de confusão por proximidade de jogadores. Os jogadores próximos fazem com que os rastreadores se confundam e as trajetórias sejam fundidas.

Em nosso caso, a observação não faz distinção entre os jogadores detectados e, portanto, o funcionamento do filtro fica prejudicado nesses casos. Isso se deve ao fato de que a medição das partículas observa apenas a probabilidade dada pela função de observação proposta na Seção 4.2.3. Picos próximos com movimento muito parecido causam divisão do conjunto de partículas e, nesse caso, a predição do sistema não consegue resolver a confusão.

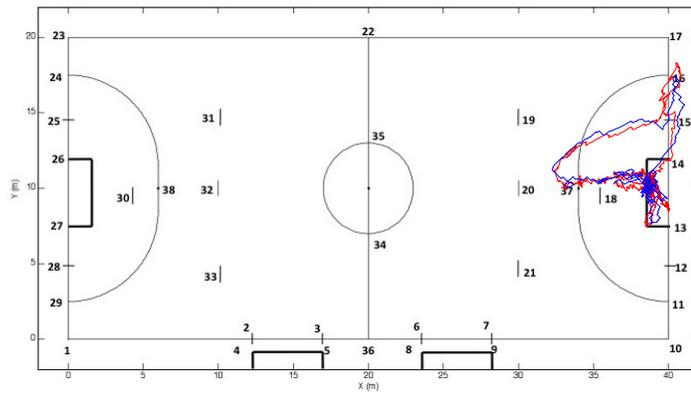
Em geral, os árbitros se posicionam na lateral da quadra e possuem dinâmica simplificada com movimentos sem muita interferência dando origem a rastreamentos bem sucedidos. Na Figura 4.6 observamos entre os quadros 134 e 169 um árbitro identificado como *Tr11* envolvido em um momento de confusão resolvido pela dinâmica do sistema.



(a) Trajetória com Maior erro



(b) Trajetória com Erro médio



(c) Trajetória com Menor erro

Figura 4.7: Trajetórias com erro médio, maior e menor encontradas pelo rastreador e marcação correspondente. As trajetórias na figura deram origem ao maior erro (4.7(a)), erro médio (4.7(b)) e menor erro (4.7(c)). A trajetória em azul corresponde à marcação manual e a trajetória em vermelho corresponde ao trajeto encontrado pelo rastreador.

A Figura 4.8 mostra os erros médios encontrados em cada quadro da trajetória com o maior erro médio, menor erro médio e a trajetória com erro médio intermediário, considerando o conjunto das trajetórias bem sucedidas. A Figura 4.7 mostra as respectivas trajetórias encontradas pelo rastreamento, plotadas juntamente com as marcações manuais. A medição do erro nas estimativas de posição encontradas pelo rastreador foram computadas comparando com as posições anotadas manualmente. A distância Euclidiana entre a posição anotada e a posição estimada nos dá o erro da estimativa.

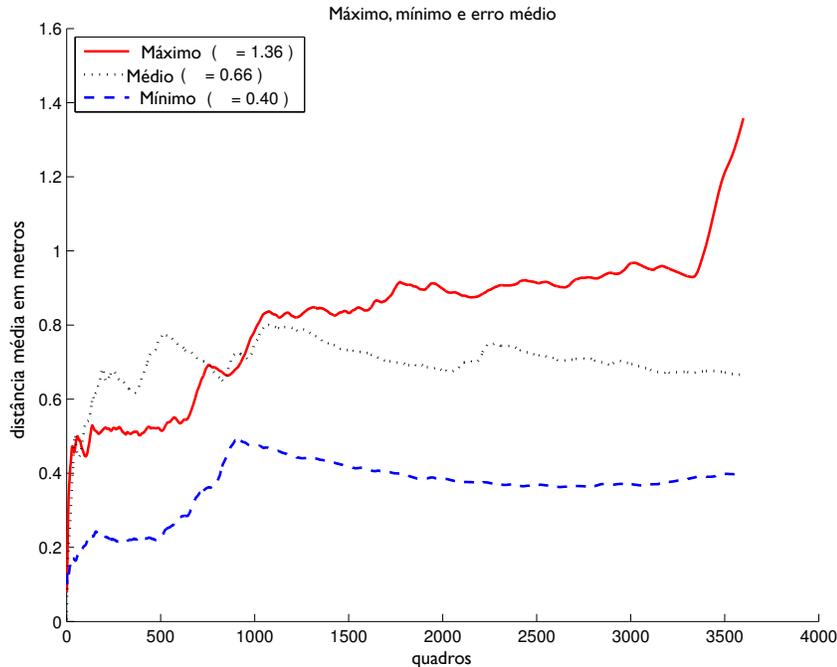


Figura 4.8: Exemplos de erros médios encontrados em trajetórias bem sucedidas. A figura apresenta as curvas encontradas para as trajetórias de maior, menor e erro médio, ambos erros acumulados.

Observando os casos bem sucedidos, notamos que o erro médio acumulado em cada trajetória fica abaixo de um metro na maioria dos casos chegando a 0,4 metros no melhor caso. Quando analisamos todas as trajetórias bem sucedidas em conjunto, encontramos um erro médio global de 0,73 metros. Um valor bastante interessante considerando as dimensões da quadra (20 metros  $\times$  40 metros) e o período de jogo (20 minutos). Esse resultado é animador dado que é obtido de forma automática após um treinamento simples nos detectores a partir de apenas um vídeo. A Tabela 4.1 mostra os erros e desvios encontrados analisando a coleção de trajetórias bem sucedidas.

Esse problema pode ser resolvido fortalecendo a observação com modelos de aparência adaptáveis, acoplados às partículas do filtro. Nessa abordagem, a partícula não apenas

Tabela 4.1: Erros médios ( $\mu$ ) e desvios ( $\sigma$ ) encontrados para as trajetórias completas encontradas pelo rastreador. Nas primeiras três colunas da tabela, encontramos a identificação do trajeto considerado.

<b>Jogo</b>	<b>Tempo</b>	<b>ID do rastreador</b>	<b><math>\mu</math></b>	<b><math>\sigma</math></b>
BoliviaxColombia	T1	player-10	0.60	0.11
BoliviaxColombia	T2	player-10	0.82	0.27
BrasilxColombia	T2	player-11	0.54	0.12
BrasilxColombia	T2	player-12	0.73	0.36
BrasilxPeru	T1	player-09	0.69	0.17
BrasilxVenezuela	T1	player-11	0.59	0.17
ColombiaxUruguai	T1	player-09	0.40	0.08
ColombiaxUruguai	T1	player-11	1.36	2.79
ColombiaxUruguai	T2	player-10	0.53	0.06
PeruxBolivia	T1	player-10	0.66	0.22
PeruxBolivia	T1	player-11	0.61	0.35
PeruxBolivia	T2	player-10	1.20	0.24
<b>Global</b>			<b>0.73</b>	<b>0.48</b>

verifica a probabilidade de existir um jogador em uma dada posição, mas verifica a probabilidade de ocorrer um jogador parecido com seu modelo de aparência. Para manter a representatividade, o modelo precisa ser atualizado ao longo do processo, já que existem variações de luz, rotação e escala do objeto observado.



# Capítulo 5

## Modelo de Aparência

No Capítulo 3, apresentamos um método para localização dos jogadores de futebol de salão no plano da quadra e, no Capítulo 4, apresentamos uma estratégia para o rastreamento dos jogadores usando a técnica apresentada no Capítulo 3 como modelo de observação. O rastreamento usando a função multimodal como observação permitiu rastrear os jogadores automaticamente. Entretanto, características da função multimodal levam o sistema a momentos que denominamos “momentos de confusão” que ocorrem em situações como por exemplo, as disputas por bola e comemorações.

A função de observação apresentada no Capítulo 3, não faz distinção entre os jogadores e, por isso, os jogadores próximos uns dos outros tendem a ser observados como apenas um. No momento em que o filtro de partículas faz a ponderação das partículas com base em sua observação, partículas que não representam bem o objeto rastreado recebem pesos semelhantes a outras partículas que representam o objeto corretamente. Isso acontece porque as partículas ruins têm o peso influenciado pela presença de um objeto diferente muito próximo do objeto rastreado.

É preciso um mecanismo que, associado à função multimodal, produza pesos maiores para partículas que realmente representam o objeto rastreado e pesos menores para partículas influenciadas por um objeto vizinho.

Durante a criação da função multimodal que representa a localização dos jogadores no plano da quadra, são usadas Gaussianas que se combinam em uma mistura compondo uma função mais complexa. Nessa mistura, todas as Gaussianas possuem o mesmo peso. A idéia principal neste capítulo é encontrar pesos diferentes para cada uma das Gaussianas da mistura, com base em um modelo de aparência.

Cada Gaussiana da mistura é resultado de uma detecção no plano da imagem. Essa detecção pode ser usada como comparação junto a um modelo de aparência conhecido e, assim, encontrar um peso diferente e proporcional a uma similaridade usada como ponderação das Gaussianas na mistura.

Usando apenas a função multimodal do Capítulo 3 como observação do filtro de partículas, é possível rastrear objetos com pouca interação entre si, dado que a função multimodal é projetada para encontrar as regiões da quadra com maior probabilidade de conter jogadores. Com um modelo de aparência associado à função multimodal, é possível encontrar as regiões com maior probabilidade de ser encontrado um jogador que seja parecido com um modelo conhecido. A função terá picos mais altos na região do jogador mais parecido com o modelo associado e picos menores para os demais, resultando em uma função de observação fortalecida.

Nas próximas seções, mostramos detalhes da implementação e também nossos experimentos usando o modelo de observação em conjunto com a estratégia de rastreamento usada no Capítulo 4.

## 5.1 Aparência do Objeto

O Capítulo 3 mostra como os jogadores são localizados no plano das imagens usando um método de detecção treinado. O resultado do detector é um conjunto de retângulos, cada um delimitando um jogador detectado. Cada retângulo é um recorte da imagem.

É preciso modelar a aparência de um recorte. Para isso, optamos por usar uma combinação de gradientes, ângulos e cores compiladas em um conjunto de histogramas. A informação de cores é importante para diferenciar recortes referentes a jogadores de times diferentes. Já ângulos e gradientes ajudam a distinguir ocorrências de jogadores diferentes mas do mesmo time considerando que estarão em posições diferentes e em poses diferentes, os contornos dentro do recorte serão diferentes.

Sabemos que a forma dos jogadores não se altera significativamente entre os quadros adjacentes do vídeo. Por esse motivo, os histogramas dos gradientes serão muito similares neste caso e podem ser uma boa representação da distribuição dos contornos ou bordas de objetos presentes no recorte detectado.

Cada recorte é dividido em três regiões diferentes: topo, meio e base. Uma aparência é composta por cinco histogramas por região, computados em seis passos:

1. Aplicamos o filtro de Sobel ao recorte para encontrar as imagens com as derivadas nas direções  $x$  e  $y$ ,  $dx$  e  $dy$  respectivamente. Para isso, filtramos a imagem de intensidade correspondente ao recorte usando os kernels  $[-1, 0, 1]$  e  $[-1, 0, 1]^T$ ;
2. Usamos  $dx$  e  $dy$  para calcular a imagem com as magnitudes dos gradientes:

$$IGr_{i,j} = \sqrt{dx_{i,j}^2 + dy_{i,j}^2}; \quad (5.1)$$

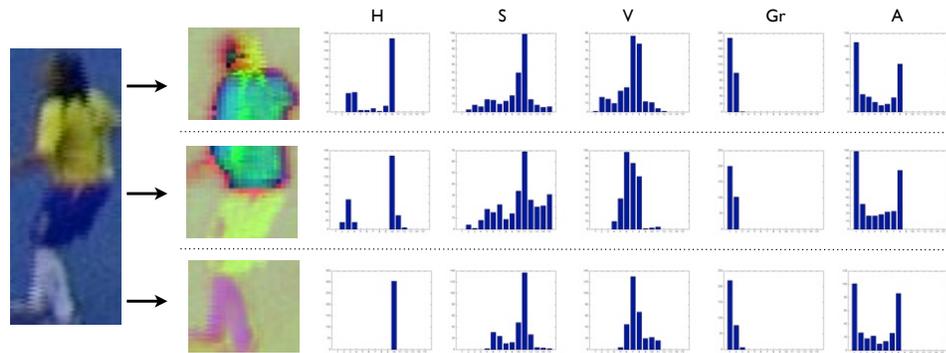


Figura 5.1: Modelo de aparência para um jogador observado. Cada recorte de imagem é convertido para HSV e dividida em três regiões: topo, meio e base. Cada região resulta em um conjunto de três histogramas por canal (H, S e V). Além disso, cada região recebe mais dois histogramas: um para gradientes (Gr) e outro para os ângulos (A).

3. Para calcular uma imagem com as orientações dos ângulos:

$$IA_{i,j} = atang \left( \frac{dy_{i,j}}{dx_{i,j}} \right); \quad (5.2)$$

4. Computamos os histogramas normalizados  $Gr$  usando  $IGr$  para as magnitudes dos gradientes e o histograma  $A$  para os ângulos;
5. Cada subimagem descrita usando o espaço de cores RGB é convertida para o espaço de cores HSV. Para cada canal de cor computamos os histogramas H, S e V respectivamente.
6. Normalizamos os histogramas H, S e V.

Dessa forma, uma aparência é composta por uma lista de 15 Histogramas. A Figura 5.1 mostra o conjunto de histogramas que compõem a aparência calculada para um recorte observado.

Para comparar dois histogramas, usamos a correlação entre histogramas. O valor um representa maior similaridade e o valor zero representa a menor. Considerando uma aparência  $AP$  como uma lista de 15 Histogramas e  $AP^i$  como o  $i$ -ésimo histograma na lista, a comparação entre duas aparências,  $AP_1$  e  $AP_2$  pode ser dada pela Equação 5.3, que define a similaridade entre as duas aparências consideradas.

$$S(AP_1, AP_2) = \prod_i^{15} CORR(AP_1^i, AP_2^i). \quad (5.3)$$

Além disso, o jogo é gravado por múltiplos pontos de vista simultâneos, que resultam em cores e ângulos diferentes entre si. Por esse motivo, é preciso a criação de modelos de aparência específicos para cada uma das câmeras. Nesse sentido, um jogador sendo rastreado precisa ser associado a quatro modelos distintos, cada um referente a uma das câmeras de nosso sistema.

## 5.2 O Modelo na Câmera

Durante o jogo, os jogadores mudam sua aparência constantemente. Ora são vistos de frente, ora são vistos de costas, mudam sua pose, correm, andam, etc. Essa grande variedade nos leva à necessidade de um modelo capaz de se adaptar durante o tempo. Para isso, usamos um modelo dinâmico que guarda uma pequena coleção ou histórico de aparências consideradas boas. O modelo adapta sua coleção às variações durante o rastreamento, incluindo novas aparências e retirando aparências antigas. Com isso, nosso modelo de aparência em cada câmera é constituído de uma pequena lista adaptável composta de aparências consideradas boas.

Cada jogador detectado possui uma aparência calculada usando o recorte da imagem original. Essa aparência é comparada com o modelo ou lista criada e mantido para a respectiva câmera. A comparação gera um coeficiente  $\pi$  que irá representar a importância da Gaussiana correspondente representar a posição do jogador que está sendo rastreado.

Dessa forma, o coeficiente  $\pi$  pode ser calculado como a maior similaridade entre a aparência detectada no quadro atual ( $AP_d$ ) e cada uma das aparências do modelo ( $AP_m$ ), tal como na Equação 5.4:

$$\pi = \max\{S(AP_d, AP_m)\}. \quad (5.4)$$

Este modelo precisa ser atualizado periodicamente para manter uma boa representação do objeto. A atualização é feita substituindo a aparência do modelo considerada um candidato a sair (CS) por uma nova, considerada um candidato a entrar (CE).

A cada comparação realizada entre o modelo e uma aparência qualquer, o histórico de utilização interno sofre uma atualização para sinalizar qual parte da lista de aparência não se usa a mais tempo. A aparência menos usada no modelo será usada como candidata a sair (CS) do modelo.

Por outro lado, uma aparência observada é considerada boa para atualizar o modelo (CE) quando gera o melhor peso usando o modelo atual. Assim como são mantidas informações de uso das aparências usadas no modelo, o sistema usa a aparência detectada que gerou o maior peso durante a observação do filtro de partículas como candidato a entrar (CE) no modelo. No caso de a aparência CE tiver similaridade maior que um

limiar a aparência CS será substituída pela aparência CE. Essa informação é mantida durante o passo de observação do filtro de partículas.

Com isso, temos um modelo de aparência acoplado à câmera, capaz de verificar a similaridade entre uma detecção e um pequeno conhecimento da aparência do objeto composto por um histórico de boas aparências.

## 5.3 Função Multimodal Fortalecida

Cada Gaussiana na mistura que compõe a função multimodal é gerada por um retângulo detectado no plano da imagem de uma das câmeras. O recorte correspondente é usado para calcular a aparência do objeto detectado. Essa aparência é comparada a um modelo de aparência conhecido para se ter uma medida de similaridade entre a aparência detectada e a aparência esperada. O resultado é um valor que pode ser usado para ponderar a Gaussiana correspondente ao retângulo gerador e, com isso, produzir uma função multimodal mais forte que aquela usando apenas as Gaussianas.

No Capítulo 3, mostramos a construção de uma função multimodal que é usada como observação no filtro de partículas do Capítulo 4. A função multimodal é composta por uma mistura de Gaussianas resultantes das projeções das detecções no plano das imagens.

Cada Gaussiana  $\mathcal{N}(X|\mu_k, \Sigma_k)$  é considerada uma componente de uma mistura que forma uma função complexa que representa a ocupação do plano de jogo. Na construção da função multimodal, temos um conjunto de detecções de mesma importância e, portanto, o mesmo coeficiente  $\psi$  que respeita as condições  $\sum \psi_k = 1$  e  $\psi = 1/K$ . Nesse sentido, a função multimodal ou o modelo de ocupação do plano de jogo pode ser expresso por:

$$M(X) = \sum_{k=1}^K \frac{1}{K} \mathcal{N}(X|\mu_k, \Sigma_k), \quad (5.5)$$

sendo  $x$  um vetor com as coordenadas  $(x, y)$  de uma posição do plano de jogo,  $K$  o número de Gaussianas da mistura e  $\mu_k$  e  $\Sigma_k$  os parâmetros conhecidos da Gaussiana, média e covariância, encontrados tal como descrito no Capítulo 3.

Usando a aparência da detecção comparada ao modelo do objeto na câmera, encontramos, para cada detecção, um coeficiente  $\pi$  dado pela Equação 5.4. Esse coeficiente representa a similaridade entre a detecção correspondente e o modelo da câmera. Podemos usá-lo como importância na mistura e re-escrever a Equação 5.5 como:

$$M(X) = \sum_{k=1}^K \pi_k \mathcal{N}(X|\mu_k, \Sigma_k). \quad (5.6)$$

A cada novo recorte encontrado, uma nova aparência é calculada. Essa nova aparência é comparada a cada aparência do modelo na câmera correspondente para encontrar a

aparência armazenada que possui maior similaridade (mais parecida dentro do modelo). A similaridade  $\pi$  encontrada é usada como peso da Gaussiana correspondente na mistura de Gaussianas que gera a função multimodal para a localização dos jogadores. Com isso, a função multimodal descrita no Capítulo 3 pode representar o potencial de encontrar um jogador em uma posição da quadra que se pareça com um modelo conhecido.

A cada nova iteração do filtro, um novo conjunto de partículas precisa ser ponderado. Para isso, as Gaussianas da mistura correspondente à função multimodal são ponderadas usando a aparência de seus correspondentes recortes. A aparência do recorte de cada Gaussiana é comparada ao modelo de aparência da câmera correspondente, mantido pelo rastreador. Dessa forma, o modelo de observação usado pelo filtro mantém  $N$  modelos de aparência referentes ao objeto observado, um para cada câmera. O resultado é uma função multimodal com picos mais altos nas regiões correspondentes ao objeto rastreado e portanto, capaz de identificar proximidade de jogadores mas distinguindo objetos diferentes.

A Figura 5.2 ilustra esse resultado. Na Figura, podemos observar dois jogadores muito próximos. Supondo um rastreador perseguindo o jogador de traje escuro, sem o uso do modelo de aparência, a tendência do sistema seria encontrar pesos semelhantes para as partículas corretas e para as partículas que indicarem o outro jogador.

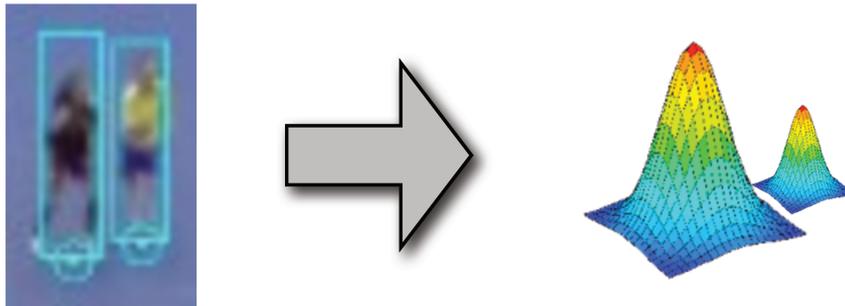


Figura 5.2: Gaussianas com pesos diferentes. Com o modelo de aparência os jogadores próximos são observados com pesos diferentes para as respectivas Gaussianas.

Com o modelo de Aparência, são gerados pesos diferentes para as respectivas Gaussianas e, portanto, as partículas referentes ao jogador correto terão maior peso no sistema de rastreamento.

## 5.4 Metodologia e Experimentos

O Capítulo 4 apresenta uma estratégia de rastreamento que utiliza como modelo de observação a função multimodal construída no Capítulo 3. A estratégia apresentada não consegue lidar com situações específicas nas quais os jogadores são confundidos pelo rastreador. Esses momentos foram chamados de “momentos de confusão” e não podem ser tratados apenas com a função multimodal.

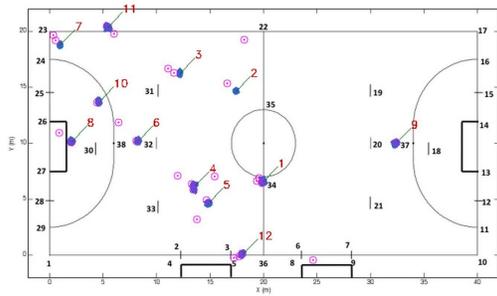
Apresentamos neste Capítulo um modelo de aparência projetado para ser usado em conjunto com a função multimodal do Capítulo 3, e assim constituir um modelo de observação fortalecido para o filtro de partículas. A aparência de uma detecção é comparada a um modelo de aparência mantido pelo rastreador e o valor de similaridade encontrado é usado como peso da Gaussiana correspondente na função multimodal. O resultado é uma função de observação capaz de fazer uma melhor distinção entre os jogadores no plano da quadra (Figura 5.2).

A configuração das câmeras utilizadas para os novos testes permanece a mesma apresentada no Capítulo 4, constituída de quatro câmeras dispostas ao redor da quadra e com alguma sobreposição entre as áreas monitoradas. Os dois primeiros minutos de cada jogo foram anotados manualmente para que tivéssemos um *ground-truth* para as comparações.

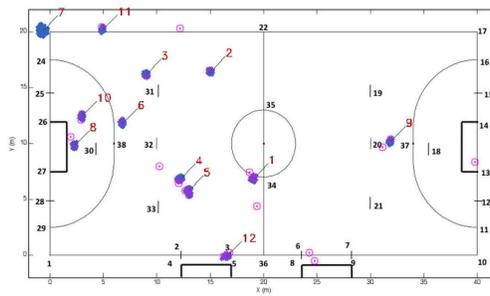
Relembramos que estamos interessados em rastrear os 10 jogadores em quadra e os dois árbitros que se encontram nas laterais da quadra. Para isso, iniciamos um rastreador independente para cada objeto de interesse no plano de jogo com 500 partículas cada um. A Figura 4.5 mostra o momento após a inicialização dos rastreadores.

Existem duas situações de confusão que afetam o rastreamento e podem resultar em trajetórias inadequadas sem o modelo de aparência. A primeira delas se refere ao momento em que dois jogadores passam um pelo outro durante o jogo e portanto geram trajetórias cruzadas. Essas situações são menos críticas e são resolvidas em sua grande maioria apenas pela predição do filtro. Na Figura 4.6 encontramos um exemplo. Entre os quadros 134 e 169 observamos que os rastreadores  $Tr02$  e  $Tr11$  continuam rastreando corretamente mesmo após um período de grande proximidade.

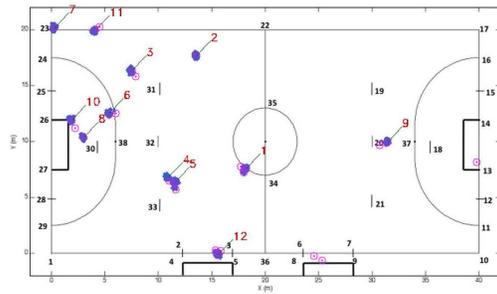
A outra situação ocorre quando dois ou mais jogadores possuem trajetórias que tendem a ser coincidentes como em uma comemoração. Nesses casos, os jogadores estão muito próximos um do outro e se movem em direções parecidas. A função de observação não consegue distinguir os jogadores envolvidos nessas situações. O resultado é a fusão de trajetórias e a perda de algum jogador envolvido. As trajetórias coincidentes também podem ocorrer em outras situações de proximidade como disputa pela bola por exemplo. A Figura 4.6 mostra dois jogadores próximos, identificados como  $Tr04$  e  $Tr05$ , cujas trajetórias se fundem após seu distanciamento. Nesse momento os respectivos rastreadores são confundidos pelo modelo de observação que não é capaz de distinguir os jogadores.



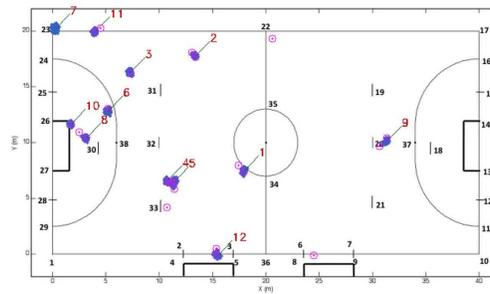
(a) quadro 007



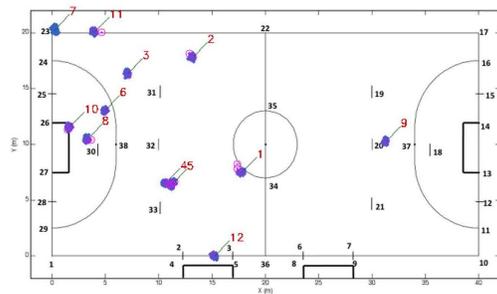
(b) quadro 030



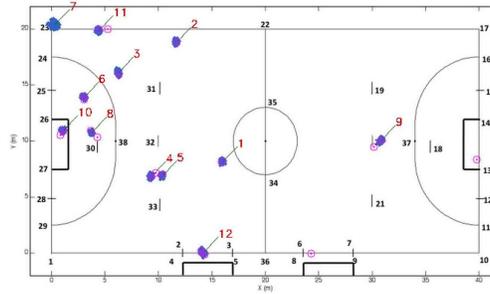
(c) quadro 049



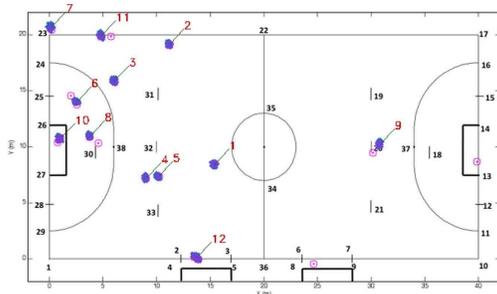
(d) quadro 052



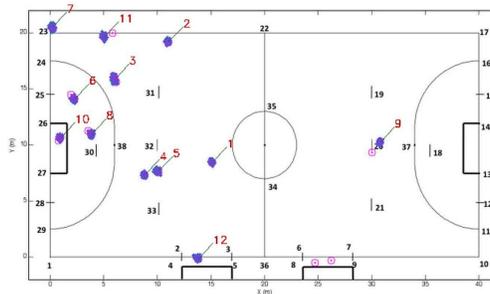
(e) quadro 055



(f) quadro 078

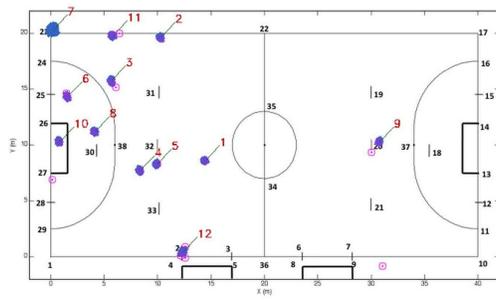


(g) quadro 085

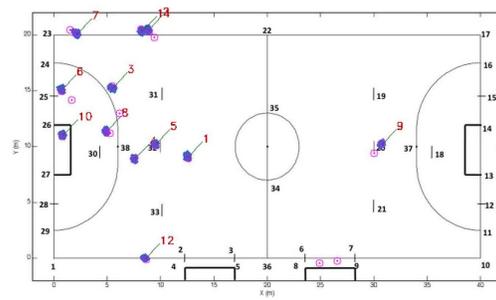


(h) quadro 089

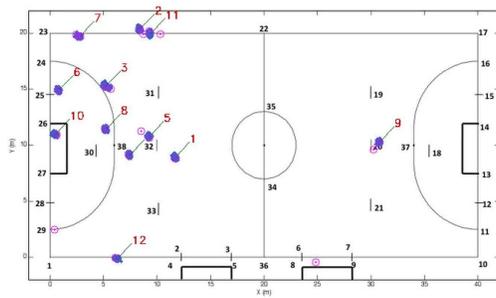
Figura 5.3: Demonstração do rastreamento com o modelo de aparência. Na Figura é possível observar que o rastreamento é mantido ao longo dos quadros do vídeo de forma mais consistente que na Figura 4.6. Foram escolhidos os mesmos quadros apresentados na Figura 4.6 para que fosse possível a comparação entre as figuras.



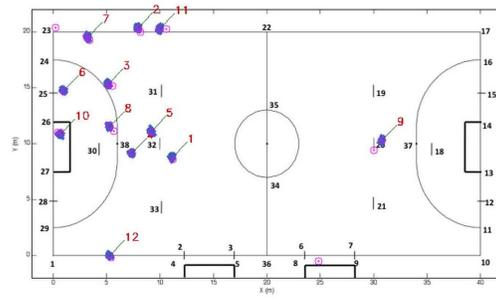
(a) quadro 100



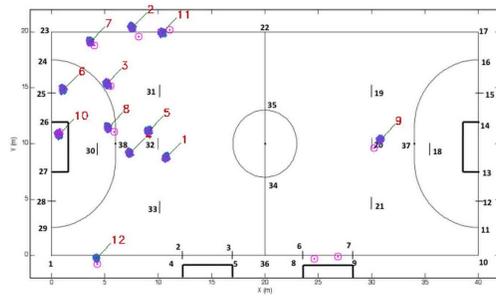
(b) quadro 134



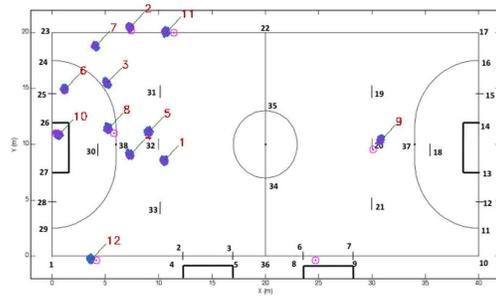
(c) quadro 151



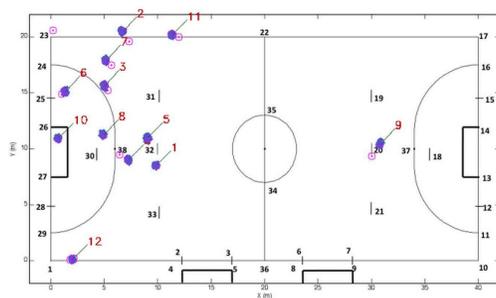
(d) quadro 161



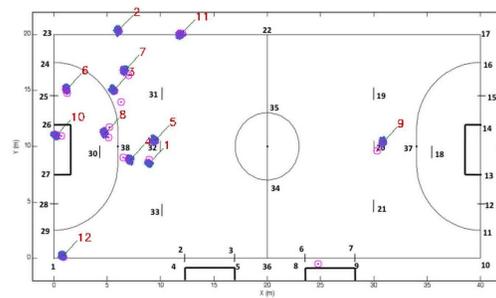
(e) quadro 169



(f) quadro 175



(g) quadro 190



(h) quadro 209

Continuação da figura 5.3.

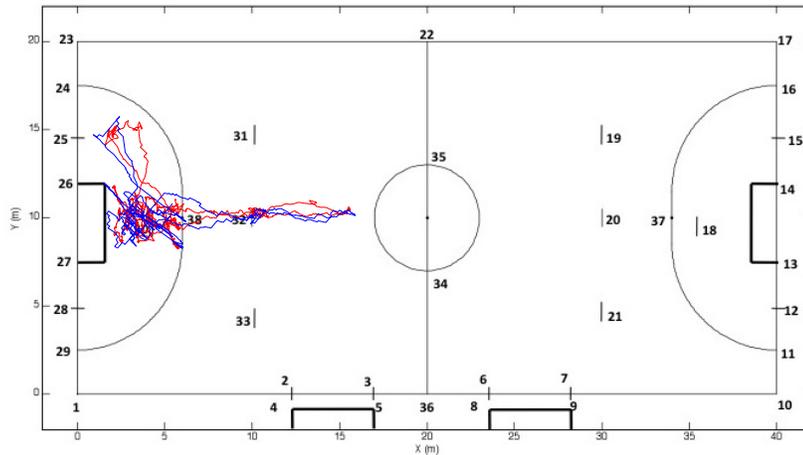


Figura 5.3: Trajetória encontrada pelo rastreador sem o modelo de aparência (vermelho) e a marcação manual correspondente (azul). A trajetória em vermelho corresponde ao erro médio encontrado na Figura 5.5.

Este problema pode ser melhor tratado usando nosso modelo de aparência adaptativo em conjunto com a função multimodal para gerar uma função de observação com pesos diferentes para Gaussianas diferentes. Nessa abordagem, as partículas recebem a probabilidade de representar, em uma dada posição, um jogador que seja parecido com um modelo. Para manter a representatividade, o modelo é atualizado periodicamente para lidar com variações de luz, rotação e escala dos objetos de interesse no plano da imagem.

### 5.4.1 Experimentos Usando do Modelo de Aparência

Usando o modelo de aparência, podemos encontrar trajetórias com erros e covariâncias ainda menores, já que menos momentos de confusão irão ocorrer.

A proximidade de jogadores de times diferentes passa a não causar confusão entre os rastreadores, quando usamos o modelo de aparência. Os quadros apresentados na Figura 4.6 são reexibidos na Figura 5.3, agora usando o modelo de aparência. Como pode ser notado na Figura 5.3, os jogadores (*Tr04* e *Tr05*) passam a ser rastreados corretamente.

A Figura 5.4 mostra a melhor trajetória encontrada usando o modelo de observação composto pela combinação do modelo de aparência com a função multimodal.

Os resultados nos mostram uma melhoria no rastreamento em consequência do uso do modelo de aparência, o que pode ser verificado comparando-se as Tabelas 5.1 e 5.2.

No primeiro caso, sem o uso do modelo de aparência, encontramos 0,73 metros de erro médio acumulado global e um desvio de 0,48, enquanto que usando o modelo de aparência no segundo caso encontramos 0,60 metros de erro médio acumulado e desvio de 0,19.

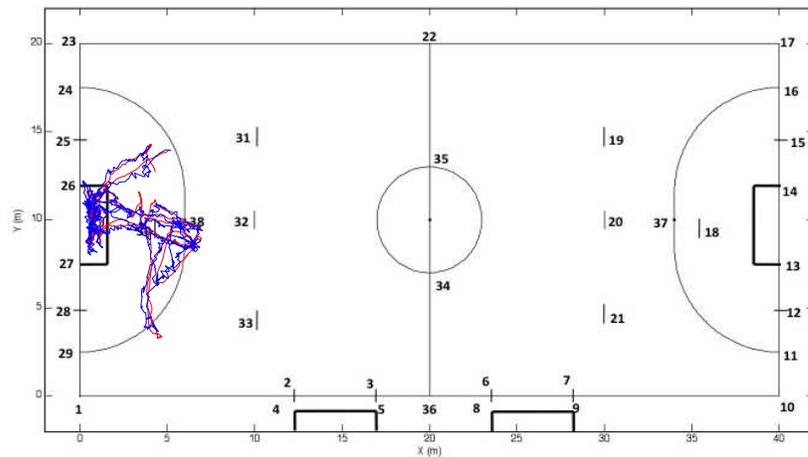


Figura 5.4: Trajetória encontrada pelo rastreador usando o modelo de aparência (vermelho) e a marcação manual correspondente (azul). A trajetória em vermelho corresponde ao menor erro encontrado na Figura 5.6.

Os gráficos com os erros médios acumulados no melhor caso, pior caso e caso médio para as duas tabelas são mostrados nas Figuras 5.5 e 5.6 respectivamente. Podemos observar comparando as figuras que no segundo caso a maioria das curvas ficaram abaixo de sua correspondente no modelo anterior.

Usando o modelo de aparência, o rastreador pode resolver alguns dos problemas apresentados na seção anterior. Por exemplo, quando dois jogadores estão em uma disputa pela bola, os movimentos são semelhantes e as posições são muito próximas. Entretanto, suas aparências são diferentes. Neste caso, as observações dos respectivos filtros de partículas vão observar dois objetos diferentes e podem obter a probabilidade correta para as partículas.

Outro problema acontece quando um jogador passa perto de um outro. Sem o modelo de aparência, os rastreadores serão confundidos percebendo apenas um jogador neste local. Se os jogadores estão se movendo lentamente, o modelo de observação sem o modelo de aparência pode mesclar as trajetórias. O modelo de aparência ajuda o modelo de observação a distinguir estes jogadores gerando pesos diferentes para jogadores diferentes. Com isso, o sistema pode continuar perseguindo o jogador correto e encontrar trajetórias bem sucedidas.

No entanto, a utilização dos modelos de aparência não resolve todos os problemas. Em casos de celebração, os jogadores se encontram muito próximos por um longo período de tempo e o detector no plano de imagem não pode encontrar todos os jogadores envolvidos na celebração. Nestes casos, o rastreador perde as informações específicas sobre o jogador rastreado e falha.

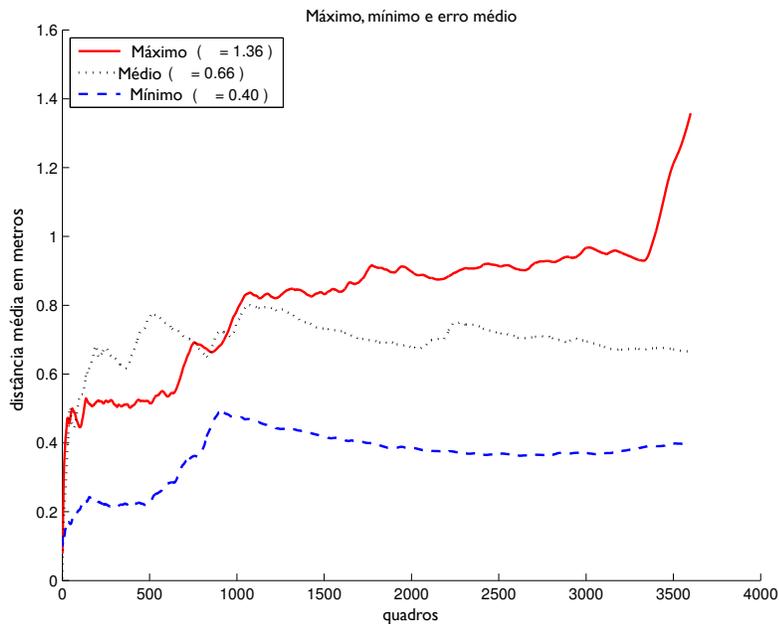


Figura 5.5: Erros médios acumulados nas trajetórias bem sucedidas considerando o rastreamento **sem o modelo de aparência**. A figura mostra as curvas encontradas para as trajetórias de erros máximo, médio e mínimo.

Tabela 5.1: Erros médios ( $\mu$ ) e desvios ( $\sigma$ ) padrão encontrados para trajetórias completas encontradas pelo rastreador **sem o modelo de aparência**. Nas três primeiras colunas da tabela, encontramos a identificação do trajeto considerado.

Jogo	Tempo	ID do rastreador	$\mu$	$\sigma$
BoliviaxColombia	T1	player-10	0.60	0.11
BoliviaxColombia	T2	player-10	0.82	0.27
BrazilxColombia	T2	player-11	0.54	0.12
BrazilxColombia	T2	player-12	0.73	0.36
BrazilxPeru	T1	player-09	0.69	0.17
BrazilxVenezuela	T1	player-11	0.59	0.17
ColombiaxUruguay	T1	player-09	0.40	0.08
ColombiaxUruguay	T1	player-11	1.36	2.79
ColombiaxUruguay	T2	player-10	0.53	0.06
PeruxBolivia	T1	player-10	0.66	0.22
PeruxBolivia	T1	player-11	0.61	0.35
PeruxBolivia	T2	player-10	1.20	0.24
<b>Global</b>			<b>0.73</b>	<b>0.48</b>

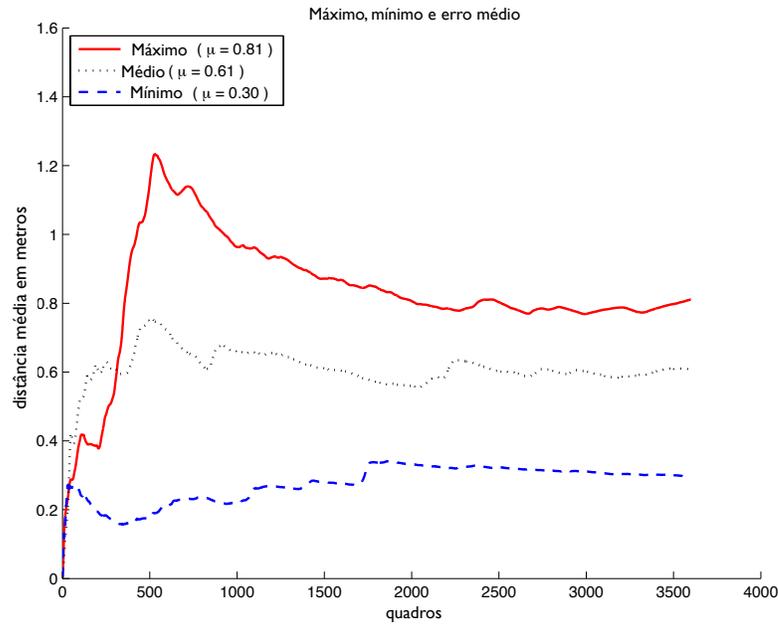


Figura 5.6: Erros médios acumulados nas trajetórias bem sucedidas considerando o rastreamento **com o modelo de aparência**. A figura mostra as curvas encontradas para as trajetórias de erros máximo, médio e mínimo.

Tabela 5.2: Erros e desvios padrão encontrados para trajetórias completas encontradas pelo rastreador **com o modelo de aparência**. Nas primeiras três colunas da tabela, encontramos a identificação do trajeto considerado.

Jogo	Tempo	ID do rastreador	$\mu$	$\sigma$
BoliviaxColombia	T1	player-10	0.57	0.11
BoliviaxColombia	T1	player-11	0.68	0.16
BoliviaxColombia	T2	player-10	0.80	0.29
BrazilxArgentina	T1	player-09	0.58	0.07
BrazilxColombia	T2	player-11	0.52	0.09
BrazilxColombia	T2	player-12	0.72	0.33
BrazilxPeru	T1	player-09	0.64	0.21
BrazilxVenezuela	T1	player-10	0.30	0.11
BrazilxVenezuela	T1	player-11	0.61	0.21
ColombiaxUruguay	T1	player-09	0.36	0.04
ColombiaxUruguay	T2	player-10	0.63	0.07
PeruxBolivia	T1	player-09	0.81	0.29
PeruxBolivia	T1	player-10	0.61	0.21
PeruxBolivia	T1	player-12	0.58	0.18
PeruxBolivia	T2	player-09	0.61	0.23
<b>Global</b>			<b>0.60</b>	<b>0.19</b>

Basicamente, existem ainda duas situações de confusão: a primeira ocorre quando dois ou mais jogadores aparecem juntos no plano da imagem. Durante o jogo, é comum encontrar jogadores de uma mesma equipe juntos comemorando um gol. Neste caso, os jogadores são fundidos em um só na função multimodal, suas aparências são muito próximas e a suas dinâmicas são as mesmas. O resultado desta situação é a fusão das trajetórias. O segundo tipo de confusão ocorre quando o detector perde um jogador que está perto de um outro com a aparência muito parecida e a mesma dinâmica de movimento. Neste caso, a função multimodal para de representar o jogador perdido e o rastreador correspondente detecta o jogador vizinho. Nesse momento o rastreador confunde o jogador vizinho com o jogador que estava sendo rastreado e então, passa a rastrear o jogador errado.

Sabemos que a intersecção das trajetórias não é problema, tanto do ponto de vista computacional quanto da análise esportiva. Entretanto, separar corretamente as trajetórias e associá-las ao jogador correto logo após os momentos de confusão não é uma tarefa fácil, principalmente considerando a grande semelhança entre os envolvidos nas confusões em momentos de comemoração. A separação das trajetórias envolvidas em comemorações durante um jogo de futsal será tema de um trabalho futuro.

# Capítulo 6

## Conclusões

Neste trabalho estivemos interessados no rastreamento de jogadores de futebol de salão durante um jogo, usando informações obtidas analisando vídeos capturados de pontos de vista diferentes e simultaneamente.

Como dados de entrada usamos apenas os vídeos capturados de pontos estratégicos e o conhecimento das coordenadas de pontos característicos da quadra de jogo. Esses pontos são organizados em um mapa e indicam a posição da marca de pênalti, o centro da quadra, os cantos e algumas intersecções entre as linhas da quadra.

Esses pontos característicos permitiram a determinação de uma matriz de transformação  $H$ , conhecida como matriz de homografia, capaz de estabelecer correspondências entre pontos no plano da imagem e pontos no plano da quadra.

O trabalho foi dividido estrategicamente em quatro partes principais, observados na Figura 1.4:

1. aquisição dos dados e sincronização do vídeos;
2. detecção dos objetos de interesse;
3. rastreamento para a extração de trajetórias e
4. modelo de aparência para fortalecer a observação.

No primeiro passo, a aquisição foi feita para câmeras estacionárias dispostas ao redor da quadra. A sincronização foi feita em laboratório usando um *software* simples, de uso livre, para edição de vídeos.

O segundo passo é dedicado à detecção e localização dos jogadores no plano da quadra, tratados no Capítulo 3. Para isso, usamos um detector treinado para localizar jogadores nas imagens. As detecções nas imagens são transformadas em aproximações dos pés dos jogadores e projetadas em um plano virtual que corresponde ao plano da quadra. As

projeções são feitas usando a matriz de homografia que corresponde à projeção da câmera onde o jogador foi detectado.

A localização dos jogadores no plano da quadra é feita por uma combinação dos dados projetados pelas diferentes câmeras. Cada ponto projetado é substituído por uma função Gaussiana com parâmetros conhecidos. O conjunto de pontos projetados dá lugar a uma mistura de Gaussianas que caracteriza a ocupação da quadra. Chamamos essa mistura de função multimodal.

Sabemos que os picos da função multimodal representam as regiões com maior probabilidade de ser encontrado um jogador. Para encontrar os picos, são utilizadas as projeções das detecções como pontos iniciais em uma subida pelo gradiente local até o pico próximo. Uma das vantagens é a eliminação de projeções redundantes vindas de câmeras diferentes. Ao final da subida, os pontos encontram o topo e são gravados em uma lista que elimina as repetições. Isso permite a verificação das posições dos jogadores com maior precisão, de forma clara e objetiva.

A validação foi feita comparando os dados estimados com os dados anotados manualmente. Com o método de localização, conseguimos encontrar posições de jogadores automaticamente com erro médio de alguns centímetros e desvios considerados pequenos. Além disso, conseguimos uma função de probabilidades que descreve a ocupação da quadra. Assim, dado um ponto podemos descobrir a probabilidade de um jogador ser encontrado naquele ponto, ideal para a fase de observação de um filtro de partículas.

O próximo passo do trabalho, tratado no Capítulo 4, se refere ao rastreamento dos jogadores. Para isso foram usados filtros de partículas com dinâmica de movimento simples e usando como observação a função multimodal.

A principal vantagem do método de rastreamento é a automação, economizando mão de obra qualificada durante todo o processo. Enquanto outras abordagens são parcialmente automatizadas ou assistidas, nosso método requer intervenção somente na fase de treinamento e no cálculo das matrizes de homografia, que são realizadas apenas uma única vez por sequência de vídeo. Por outro lado, o método ainda não representa uma solução final para rastreamento de jogadores de futebol de salão, dado que existem algumas situações que o modelo de observação não é capaz de lidar, como longos períodos de trajetórias coincidentes.

As confusões causadas por proximidade são resolvidas pela predição do rastreador nos casos em que as trajetórias se cruzam, já que a direção dos movimentos não é a mesma. Entretanto, quando as trajetórias são coincidentes, o rastreamento não consegue se ajustar dado que a observação não distingue os objetos de interesse. Essas situações podem ocorrer em uma disputa por bola ou em uma comemoração.

A quarta e última parte de nosso trabalho apresenta um modelo de aparência adaptativo (Capítulo 5). O objetivo do modelo é dar à observação do filtro de partículas a

capacidade de distinção entre os jogadores observados.

Para cada detecção no plano da imagem, uma aparência do recorte é calculada com base em um conjunto bem definido de histogramas. Com isso, cada detecção possui um recorte detectado e uma aparência detectada.

Do ponto de vista do rastreador, um objeto detectado pode ter até quatro aparências detectadas, uma para cada câmera. O rastreador mantém um modelo de aparência para cada câmera. Esse modelo guarda um pequeno conhecimento de aparências já encontradas para o objeto rastreado. Quando uma aparência detectada é comparada ao modelo correspondente, obtemos um valor de similaridade. As similaridades encontradas são usadas como pesos das Gaussianas correspondentes em uma função multimodal fortalecida.

Com a nova função de observação do rastreador, usando o modelo de aparência, alguns problemas de confusão causados por proximidade de trajetórias foram resolvidos. Em nossos testes, podemos observar que o uso de modelos de aparência em conjunto com a função multimodal permite que o filtro de partículas consiga resolver importantes situações de confusão entre os jogadores, principalmente em casos com jogadores de diferentes equipes.

O método resolve outras situações, quando os jogadores da mesma equipe são envolvidos, mas ainda observamos problemas durante as celebrações. Nestes momentos, alguns jogadores da mesma equipe aparecem juntos, e possuem dinâmica de movimento muito semelhantes que levam o detector a falhar e o rastreador a perder os jogadores envolvidos.

Outro momento de falha corre quando o detector perde o jogador rastreado em uma posição próxima de um vizinho parecido. Nesse caso, o rastreador assume o vizinho como o sendo objeto rastreado. Quando o detector volta a perceber o objeto correto, o rastreador não volta a rastrear o objeto original. Isso ocorre pela grande semelhança entre os objetos envolvidos. No contexto do futebol de salão, observamos jogadores usando uniformes com o mesmo padrão de cores e também de formas. Com isso, a semelhança das aparências envolvidas na confusão faz com que o rastreador se adapte corretamente ao objeto vizinho.

## 6.1 **Trabalhos Futuros**

A grande limitação do trabalho é o tratamento das confusões. Em um próximo passo, pretende-se concentrar no problema de confusões para fazer um modelo de observação que possa lidar com este problema. Com um modelo capaz de lidar com as situações de confusão, teremos um método completamente automático para rastrear os jogadores de futebol de salão durante todo o período de jogo.

Outra forma de tentar minimizar os momentos de confusão seria introduzir ao sistema pelo menos uma câmera equipada com lentes grande angulares e observando a quadra de jogo de cima para baixo. O objetivo é ter uma visão completa da quadra vista por cima.

Com isso um detector diferente, treinado para detectar pessoas vistas de cima ajudaria na separação da ambigüidade causada pelos momentos de confusão.

Alguns dos momentos de confusão foram causados por falha dos detectores. Se a detecção de um jogador falha por alguns instantes consecutivos quando está muito próximo de outro do mesmo time, o rastreador passa a rastrear o vizinho mas não volta para o jogador original. Apesar de utilizarmos um detector com bons resultados apresentados pela literatura, uma comparação do uso de outros detectores como parte do modelo de observação do filtro de partículas, permitiria decidir qual melhor se adequa em qual situação.

Atualmente não existe no sistema a preocupação com o tempo de execução. Entretanto, existem no código várias regiões paralelizáveis. Nossa implementação foi desenvolvida em C++ usando a biblioteca OpenCV que, atualmente, possui suporte a multi-core. A utilização massiva do paralelismo traria ganhos significativos em desempenho, principalmente levando em conta as arquiteturas multi-core existentes no mercado.

E por último, pensando na utilização do sistema por outros usuários de forma fácil e eficaz, uma *interface* gráfica poderia ser utilizada como intermediário entre o usuário e o sistema de processamento. Com isso, nosso sistema para rastreamento de jogadores de futsal ganharia visibilidade e poderia ser utilizado por pessoas não especializadas.

# Referências Bibliográficas

- [1] Fakhreddine Ababsa e Malik Malle. Robust line tracking using a particle filter for camera pose estimation. In *Proceedings of the ACM symposium on Virtual reality software and technology*, VRST '06, pages 207–211, New York, NY, USA, 2006. ACM. ISBN 1-59593-321-2. URL <http://doi.acm.org/10.1145/1180495.1180537>.
- [2] Alexandre Alahi, Yannick Boursier, Laurent Jacques, e Pierre Vanderghenst. Sport player detection and tracking with a mixed network of planar and omnidirectional cameras. *Third ACM/IEEE International Conference on Distributed Smart Cameras*, pages 7 – 14, AUG 30-SEP 02 2009.
- [3] Anjum Ali e J. K. Aggarwal. Segmentation and recognition of continuous human activity. *Detection and Recognition of Events in Video, IEEE Workshop on*, 0:28, 2001.
- [4] S. Avidan. Support vector tracking. *IEEE Trans. Pattern Anal. Mach. Intell.*, 26(8):1064–1072, August 2004. ISSN 0162-8828. URL <http://dx.doi.org/10.1109/TPAMI.2004.53>.
- [5] D. H. Ballard e C. M. Brown. *Computer Vision*. Prentice-Hall, Englewood Cliffs, NJ, 1982.
- [6] R. M. L. Barros, P. J. Figueroa, R. Anido, S. Cunha, M. Misuta, N. J. Leite, E. Lima Filho, e R. Brenzilofer. Automatic tracking of soccer players. In *XVIII Congress of the International society of Biomechanics*, pages 236–237, Zürich, 2001.
- [7] Ricardo Barros, Rafael Menezes, Tiago Russomanno, Milton Misuta, Bruno Brandão, Pascual Figueroa, Neucimar Leite, e Siome Goldenstein. Measuring handball players trajectories using an automatically trained boosting algorithm. *Computer Methods in Biomechanics and Biomedical Engineering*, 14(1):53–63, 2011.
- [8] Michael J. Black e Allan D. Jepson. Eigentracking: Robust matching and tracking of articulated objects using a view-based representation. *Int.*

- J. Comput. Vision*, 26(1):63–84, January 1998. ISSN 0920-5691. URL <http://dx.doi.org/10.1023/A:1007939232436>.
- [9] Dorin Comaniciu, Visvanathan Ramesh, e Peter Meer. Kernel-based object tracking, May 2003. ISSN 0162-8828. URL <http://dx.doi.org/10.1109/TPAMI.2003.1195991>.
- [10] Ricardo Machado Leite de Barros, Felipe Guazzibe Bergo, Ricardo Anido, Sérgio Augusto Cunha, Euclides C. Lima Filho, René Brenzicofer, e João Batista Freire. Sistema para anotação de ações de jogadores de futebol. *Revista Brasileira Ciência e Movimento*, 10:7–14, 2002.
- [11] V. Di Salvo, R. Baron, H. Tschan, F.J. Calderon Montero, N. Bachl, e F. Pigozzi. Performance characteristics according to playing position in elite soccer. *International Journal of Sports Medicine*, March 2007.
- [12] V. Di Salvo, A. Collins, B. Mc Neill, e M. Cardinale. Validation of prozone ®: A new video-based performance analysis system. *International Journal of Performance Analysis in Sport*, june 2006.
- [13] G. J. Edwards, C. J. Taylor, e T. F. Cootes. Interpreting face images using active appearance models. In *Proceedings of the 3rd. International Conference on Face & Gesture Recognition*, FG '98, pages 300–, Washington, DC, USA, 1998. IEEE Computer Society. ISBN 0-8186-8344-9. URL <http://dl.acm.org/citation.cfm?id=520809.796067>.
- [14] Pedro F. Felzenszwalb, Ross B. Girshick, David McAllester, e Deva Ramanan. Object detection with discriminatively trained part-based models. *IEEE Trans. Pattern Anal. Mach. Intell.*, 32(9):1627–1645, sep 2010. ISSN 0162-8828. URL <http://dx.doi.org/10.1109/TPAMI.2009.167>.
- [15] Paul Fieguth e Demetri Terzopoulos. Color-based tracking of heads and other mobile objects at video frame rates. In *Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition (CVPR '97)*, CVPR '97, pages 21–, Washington, DC, USA, 1997. IEEE Computer Society. ISBN 0-8186-7822-4. URL <http://dl.acm.org/citation.cfm?id=794189.794360>.
- [16] P. Figueroa, N. Leite, R.M.L. Barros, I. Cohen, e G. Medioni. Tracking soccer players using the graph representation. In *17th International Conference on Pattern Recognition*, ICPR'04, pages 787–790. Cambridge-UK, 2004.

- [17] Pascual Figueroa, Neucimar Leite, Ricardo M. L. Barros, Isaac Cohen, e Gerard Medioni. Tracking soccer players using the graph representation. In *17th International Conference on Pattern Recognition*, ICPR '04, pages 787–790, Washington, DC, USA, 2004. IEEE Computer Society. ISBN 0-7695-2128-2. URL <http://dx.doi.org/10.1109/ICPR.2004.950>.
- [18] Pascual J. Figueroa, Neucimar J. Leite, e Ricardo M. L. Barros. Tracking soccer players aiming their kinematical motion analysis. *Comput. Vis. Image Underst.*, 101(2):122–135, February 2006. ISSN 1077-3142. URL <http://dx.doi.org/10.1016/j.cviu.2005.07.006>.
- [19] David Forsyth e Jean Ponce. *Computer Vision: A Modern Approach*. Prentice Hall, 1ed edition, 2002.
- [20] Erikson Freitas de Moraes. Rastreamento e contagem de peixes utilizando filtro preditivo. Master's thesis, Universidade Federal de Minas Gerais, 2005.
- [21] Yoav Freund e Robert E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. In *Proceedings of the Second European Conference on Computational Learning Theory*, EuroCOLT '95, pages 23–37, London, UK, UK, 1995. Springer-Verlag. ISBN 3-540-59119-2. URL <http://dl.acm.org/citation.cfm?id=646943.712093>.
- [22] William B. Gevarter. *Robotics and Artificial Intelligence Applications Series: Overviews*. Business/Technology Books, 1984.
- [23] Siome Goldenstein. A gentle introduction to predictive filters. *Revista de Informatica Teórica e Aplicada (RITA)*, XI(1):61–89, 2004. ISSN 2175-2745.
- [24] E. Hennig e R. Briehle. Game analysis by gps satellite tracking of soccer players. In *XI Congress of the Canadian Society for Biomechanics*, page 44, Montreal, 2000.
- [25] N. Inamoto e H. Saito. Virtual viewpoint replay for a soccer match by view interpolation from multiple cameras. *Trans. Multi.*, 9(6):1155–1166, October 2007. ISSN 1520-9210. URL <http://dx.doi.org/10.1109/TMM.2007.902832>.
- [26] Michael Isard e Andrew Blake. Condensation - conditional density propagation for visual tracking. *International Journal of Computer Vision*, 29:5–28, 1998.
- [27] Michael Isard e Andrew Blake. Icondensation: Unifying low-level and high-level tracking in a stochastic framework. In *Proceedings of the 5th European*

- Conference on Computer Vision-Volume I - Volume I*, ECCV '98, pages 893–908, London, UK, UK, 1998. Springer-Verlag. ISBN 3-540-64569-1. URL <http://dl.acm.org/citation.cfm?id=645311.649045>.
- [28] Sachiko Iwase e Hideo Saito. Parallel tracking of all soccer players by integrating detected positions in multiple view images. In *17th International Conference on Pattern Recognition*, volume 4 of *ICPR '04*, pages 751–754, Washington, DC, USA, 2004. IEEE Computer Society. ISBN 0-7695-2128-2. URL <http://dx.doi.org/10.1109/ICPR.2004.675>.
- [29] Chia-Feng Juang, Wen-Kai Sun, e Guo-Cyuan Chen. Object detection by color histogram-based fuzzy classifier with support vector learning. *Neurocomput.*, 72(10-12):2464–2476, jun 2009. ISSN 0925-2312. URL <http://dx.doi.org/10.1016/j.neucom.2008.11.016>.
- [30] Jinman Kang, Isaac Cohen, e Gerard Medioni. Soccer player tracking across uncalibrated camera streams. In *5th IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance in conjunction with ICCV, VS-PETS'03*, pages 172–179, Nice, France, oct 2003.
- [31] Saad M. Khan e Mubarak Shah. A multiview approach to tracking people in crowded scenes using a planar homography constraint. In *Proceedings of the 9th European conference on Computer Vision - Volume Part IV*, ECCV'06, pages 133–146, Berlin, Heidelberg, 2006. Springer-Verlag. ISBN 3-540-33838-1, 978-3-540-33838-3. URL [http://dx.doi.org/10.1007/11744085\\_1](http://dx.doi.org/10.1007/11744085_1).
- [32] Seyed Hossein Khatoonabadi e Mohammad Rahmati. Automatic soccer players tracking in goal scenes by camera motion elimination. *Image Vision Comput.*, 27(4):469–479, March 2009. ISSN 0262-8856. URL <http://dx.doi.org/10.1016/j.imavis.2008.06.015>.
- [33] D.G. Liebermann, L. Katz, M.D. Hughes, R.M. Bartlett, J. McClements, e I. M. Franks. Advances in the application of information technology to sport performance. *Journal of Sports Sciences*, pages 755–769, Oct 2002.
- [34] Wei-Lwun Lu, Kenji Okuma, e James J. Little. Tracking and recognizing actions of multiple hockey players using the boosted particle filter. *Image Vision Comput.*, 27(1-2):189–205, January 2009. ISSN 0262-8856. URL <http://dx.doi.org/10.1016/j.imavis.2008.02.008>.
- [35] P. L. Mazzeo, P. Spagnolo, M. Leo, e T. D'Orazio. Visual players detection and tracking in soccer matches. In *Proceedings of the 2008 IEEE Fifth International*

- Conference on Advanced Video and Signal Based Surveillance*, AVSS '08, pages 326–333, Washington, DC, USA, 2008. IEEE Computer Society. ISBN 978-0-7695-3341-4. URL <http://dx.doi.org/10.1109/AVSS.2008.33>.
- [36] Zaher Merhi, Milad Ghantous, Mohammad Elgamel, Magdy Bayoumi, e Ayman El-Desouki. A fully-pipelined parallel architecture for kalman tracking filter. In *Proceedings of the 2007 IEEE International Conference on Application-Specific Systems, Architectures and Processors*, ASAP '07, pages 81–86, Washington, DC, USA, 2007. IEEE Computer Society. ISBN 978-1-4244-1026-2. URL <http://dx.doi.org/10.1109/ASAP.2007.4429968>.
- [37] Yu Ming, Cui Guodong, e Qi Lichao. Player detection algorithm based on gaussian mixture models background modeling. In *Proceedings of the 2009 Second International Conference on Intelligent Networks and Intelligent Systems*, ICINIS '09, pages 323–326, Washington, DC, USA, 2009. IEEE Computer Society. ISBN 978-0-7695-3852-5. URL <http://dx.doi.org/10.1109/ICINIS.2009.89>.
- [38] M.S. Misuta, R.P. Menezes, P.J. Figueroa, S.A. Cunha, e R.M.L. Barros. Representation and analysis of soccer players' trajectories. In *XXth Congress of the International Society of Biomechanics*, 2005.
- [39] Jun Miura e Hiroyuki Kubo. Tracking players in highly complex scenes in broadcast soccer video using a constraint satisfaction approach. In *Proceedings of the 2008 international conference on Content-based image and video retrieval*, CIVR '08, pages 505–514, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-070-8. URL <http://doi.acm.org/10.1145/1386352.1386417>.
- [40] Baback Moghaddam e Alex Pentland. Probabilistic visual learning for object representation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 19(7):696–710, July 1997. ISSN 0162-8828. URL <http://dx.doi.org/10.1109/34.598227>.
- [41] Krusturup P. Mohr, M. e J. Bangsbo. Match performance of high-standard soccer players with special reference to development of fatigue. *Journal of Sports Sciences*, pages 519–528, Jul 2003.
- [42] M. Mohr, P. Krusturup, e J. Bangsbo. Fatigue in soccer: a brief review. *Journal of Sports Sciences*, pages 593–599, Jun 2005.
- [43] Erikson F. Morais, Mario F. M. Campos, Flavio L. C. Padua, e Rodrigo L. Carceroni. Particle filter-based predictive tracking for robust fish counting. In *Proceedings of*

- the XVIII Brazilian Symposium on Computer Graphics and Image Processing, SIBGRAPI '05*, pages 367–374, Washington, DC, USA, 2005. IEEE Computer Society. ISBN 0-7695-2389-7. URL <http://dx.doi.org/10.1109/SIBGRAPI.2005.36>.
- [44] O. and Nagahama H. Ohashi, J. and Miyagi, T. Ogushi, e K. Ohashi. Application of an analysis system evaluating intermittent activity during a soccer match. In *IV Science and Football*, pages 32–136, 2002.
- [45] Constantine P. Papageorgiou, Michael Oren, e Tomaso Poggio. A general framework for object detection. In *Proceedings of the Sixth International Conference on Computer Vision, ICCV '98*, pages 555–, Washington, DC, USA, 1998. IEEE Computer Society. ISBN 81-7319-221-9. URL <http://dl.acm.org/citation.cfm?id=938978.939174>.
- [46] Nikos Paragios e Rachid Deriche. Geodesic active regions and level set methods for supervised texture segmentation. *Int. J. Comput. Vision*, 46(3):223–247, February 2002. ISSN 0920-5691. URL <http://dx.doi.org/10.1023/A:1014080923068>.
- [47] Sangho Park. A hierarchical bayesian network for event recognition of human actions and interactions. In *Association For Computing Machinery Multimedia Systems Journal*, pages 164–179, 2004.
- [48] E. Rampinini, D. Bishop, S.M. Marcora, D. Ferrari Bravo, R. Sassi, e F.M. Impellizzeri. Validity of simple field tests as indicators of match-related physical performance in top-level professional soccer players. *International Journal of Sports Medicine*, pages 228–235, March 2007.
- [49] T. Reilly e V. Thomas. A motion analysis of work-rate in different positional roles in professional football match play. *Journal of Human Movement Studies*, pages 87–97, 1976.
- [50] E. Rienzi, B. Drust, T. Reilly, J.E. Carter, e A. Martin. Investigation of anthropometric and work-rate profiles of elite. In *Sports Medicine and Physical Fitness*, pages 162–169, 2000.
- [51] David Serby e Luc Van Gool. Probabilistic object tracking using multiple features. In *IEEE International Conference of Pattern Recognition (ICPR)*, pages 184–187, 2004.
- [52] D. Setterwall. Computerised video analysis of football: technical and commercial possibilities for football coaching. Master's thesis, Stockholm University, 2003.

- [53] M. Shiokawa, K. Takahashi, A. Kan, K.O.S. Usui, C.S. Choi, e T. Deguchi. Computer analysis of a soccer game by the dlt method focusing on the movement of the players and the ball. In *V World Congress of Science and Football*, page 267, Lisbon, 2003.
- [54] Jaspreet Singh, Upamanyu Madhow, Rajesh Kumar, Subhash Suri, e Richard Cagley. Tracking multiple targets using binary proximity sensors. In *Proceedings of the 6th international conference on Information processing in sensor networks*, IPSN '07, pages 529–538, New York, NY, USA, 2007. ACM. ISBN 978-1-59593-638-7. URL <http://doi.acm.org/10.1145/1236360.1236427>.
- [55] Janez Pers Stanislav e Stanislav Kovacic. Computer vision system for tracking players in sports games. In *First Int'l Workshop on Image and Signal Processing and Analysis*, IWISPA'2000, pages 81–86, 2000.
- [56] Chris Stauffer e W. Eric L. Grimson. Adaptive background mixture models for real-time tracking. In *Computer Vision and Pattern Recognition*, CVPR'99, pages 2246–2252, Fort Colins, CO, june 1999.
- [57] S. Toki e S. Sakurai. Quantitative match analysis of soccer games with two dimensional dlt procedures. In *XXth Congress of International Society of Biomechanics*, page 911, Cleveland-USA, 2005.
- [58] Xiaofeng Tong, Jia Liu, Tao Wang, e Yimin Zhang. Automatic player labeling, tracking and field registration and trajectory mapping in broadcast soccer video. *ACM Trans. Intell. Syst. Technol.*, 2(2):15:1–15:32, feb 2011. ISSN 2157-6904. URL <http://doi.acm.org/10.1145/1899412.1899419>.
- [59] Emanuele Trucco e Alessandro Verri. *Introduction Technique for 3-D Computer Vision*. Prentice Hall, 1998.
- [60] C. J. Veenman, M.J.T. Reinders, e E. Backer. Resolving motion correspondence for densely moving points. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23:54–72, 2001.
- [61] P. Viola e M. Jones. Rapid object detection using a boosted cascade of simple features. In *Computer Vision and Pattern Recognition*, volume 1 of *CVPR 2001*, pages I–511 – I–518 vol.1, 2001.
- [62] Greg Welch e Gary Bishop. An introduction to the kalman filter. Technical report, Chapel Hill, NC, USA, 1995.

- [63] Ming Xu, Liam Lowey, e James Orwell. Architecture and algorithms for tracking football players with multiple cameras. In *Vision, Image and Signal Processing*, volume 52, pages 232–241, 2005.
- [64] Y. Yagi, Y. Nishizawa, e M. Yachida. Map-based navigation for a mobile robot with omnidirectional image sensor copis. *Robotics and Automation, IEEE Transactions on*, 11(5):634–648, oct 1995. ISSN 1042-296X.
- [65] Alper Yilmaz, Omar Javed, e Mubarak Shah. Object tracking: A survey. *ACM Comput. Surv.*, 38(4), December 2006. ISSN 0360-0300. URL <http://doi.acm.org/10.1145/1177352.1177355>.
- [66] Alper Yilmaz, Xin Li, e Mubarak Shah. Contour-based object tracking with occlusion handling in video acquired using mobile cameras. *IEEE Trans. Pattern Anal. Mach. Intell.*, 26(11):1531–1536, November 2004. ISSN 0162-8828. URL <http://dx.doi.org/10.1109/TPAMI.2004.96>.
- [67] S. C. Zhu, T. S. Lee, e A. L. Yuille. Region competition: unifying snakes, region growing, energy/bayes/mdl for multi-band image segmentation. In *Proceedings of the Fifth International Conference on Computer Vision, ICCV '95*, pages 416–, Washington, DC, USA, 1995. IEEE Computer Society. ISBN 0-8186-7042-8. URL <http://dl.acm.org/citation.cfm?id=839277.840064>.

# Apêndice A

## Gráficos dos erros médios acumulados para a localização dos jogadores no plano da quadra

No Tabela 3.2 as médias e desvios encontrados para cada uma das câmeras e em cada um dos jogos analisados. A Figura 3.8 apresenta apenas o melhor e o pior casos encontrados. Apresentamos aqui, os gráficos para cada linha da Tabela 3.2.

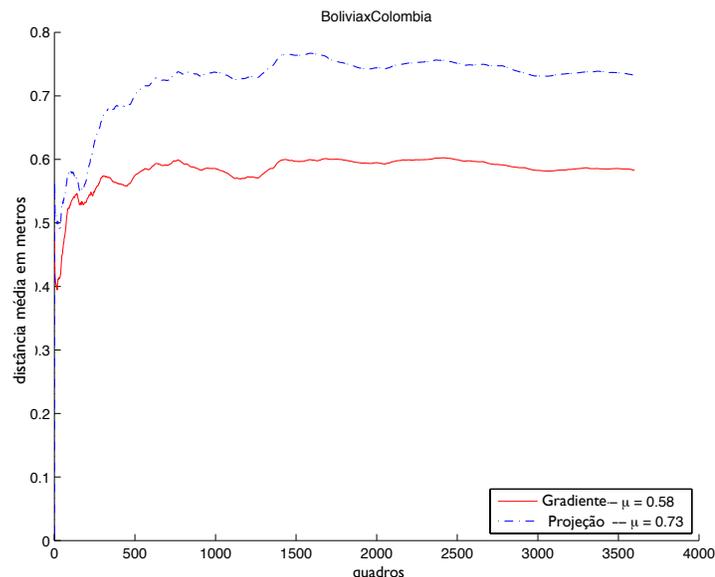


Figura A.1: Erro médio na estimativa observado na câmera 1, tempo 1 do jogo Bolívia x Colômbia.

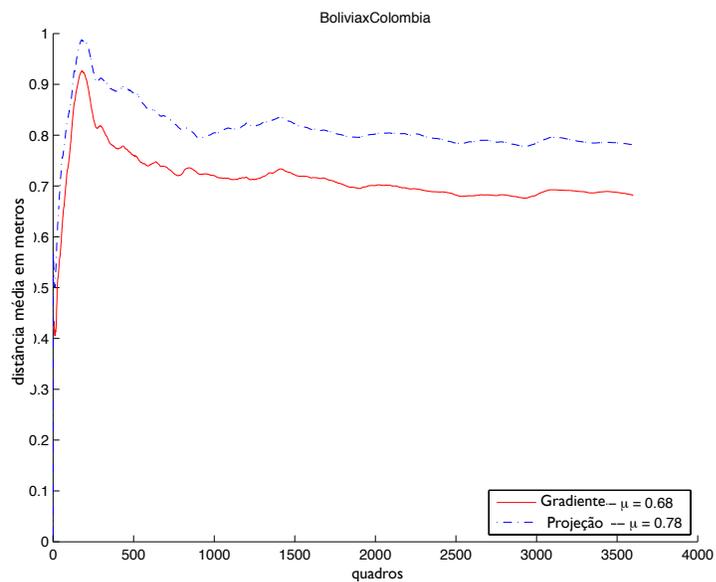


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 2, tempo 1 do jogo BoliviAxColombia.

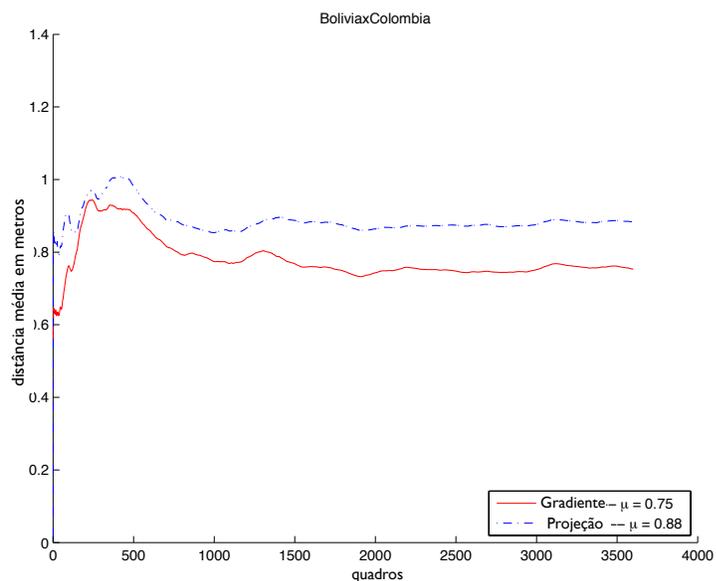


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 3, tempo 1 do jogo BoliviAxColombia.

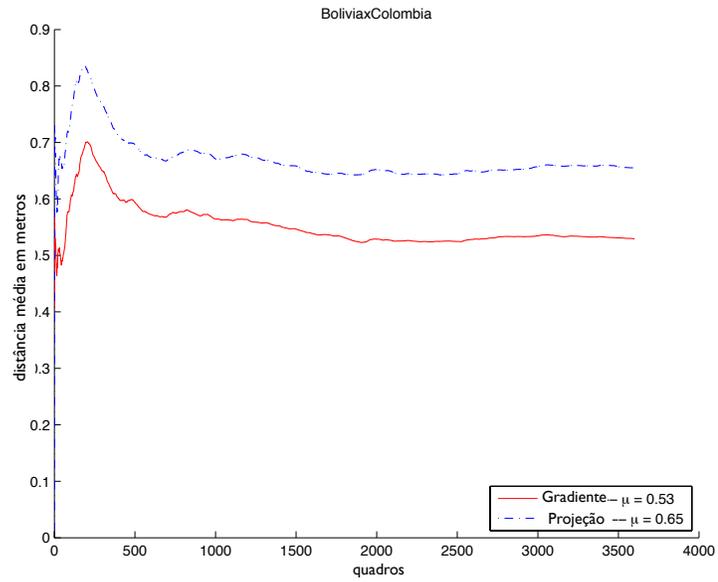


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 4, tempo 2 do jogo BoliviaColombia.

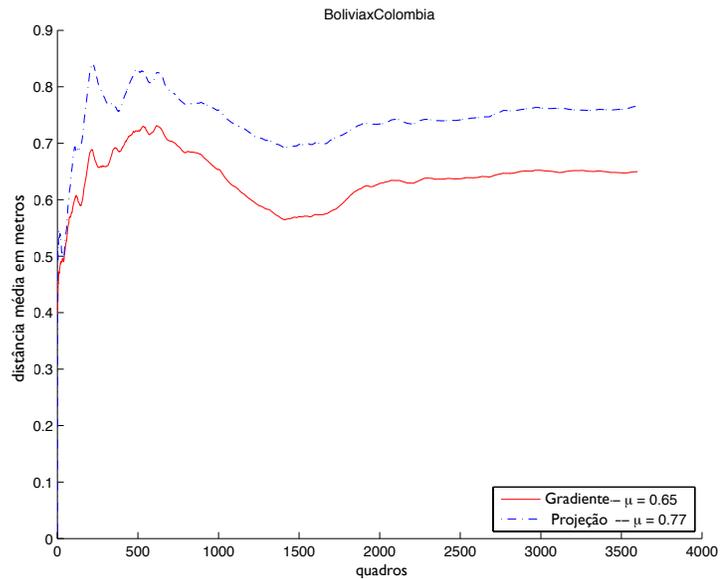


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 1, tempo 2 do jogo BoliviaColombia.

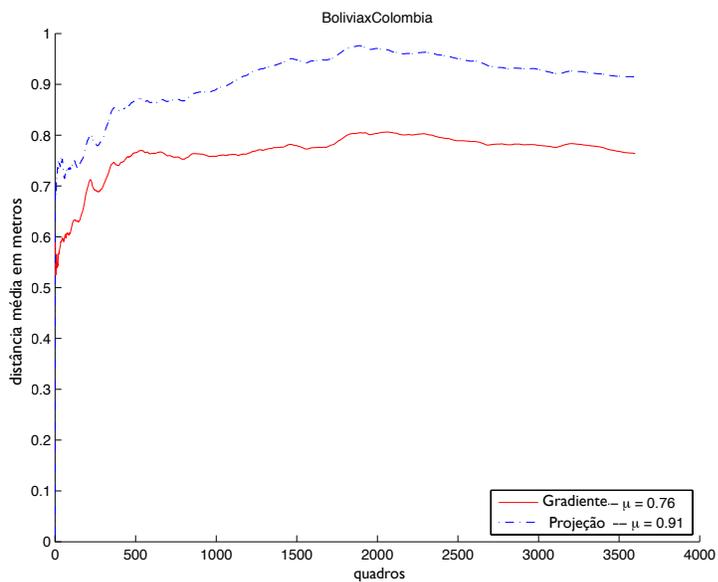


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 2, tempo 2 do jogo BoliviAxColombia.

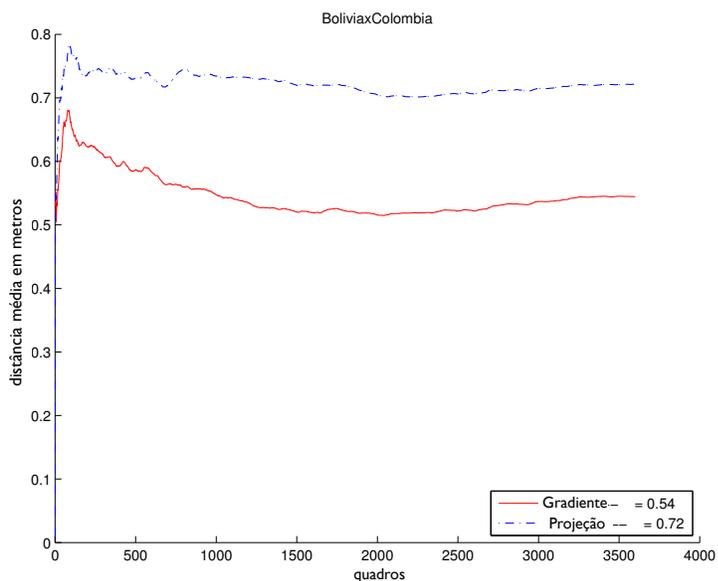


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 3, tempo 2 do jogo BoliviAxColombia.

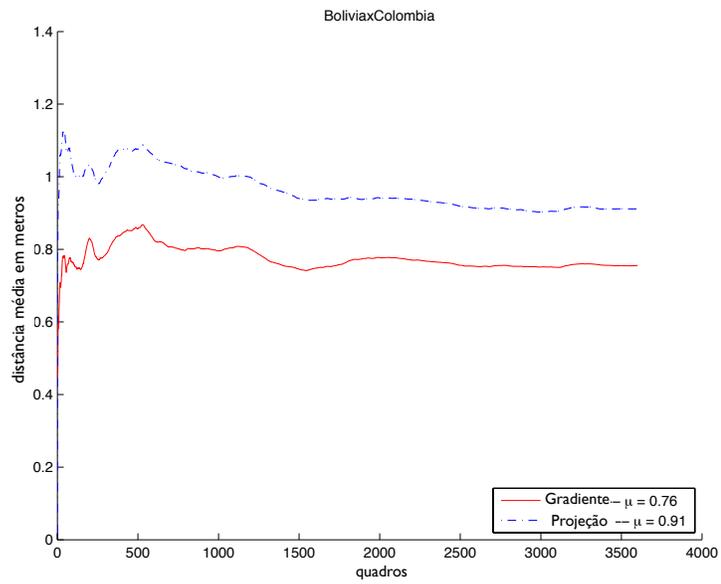


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 4, tempo 2 do jogo BoliviaColombia.

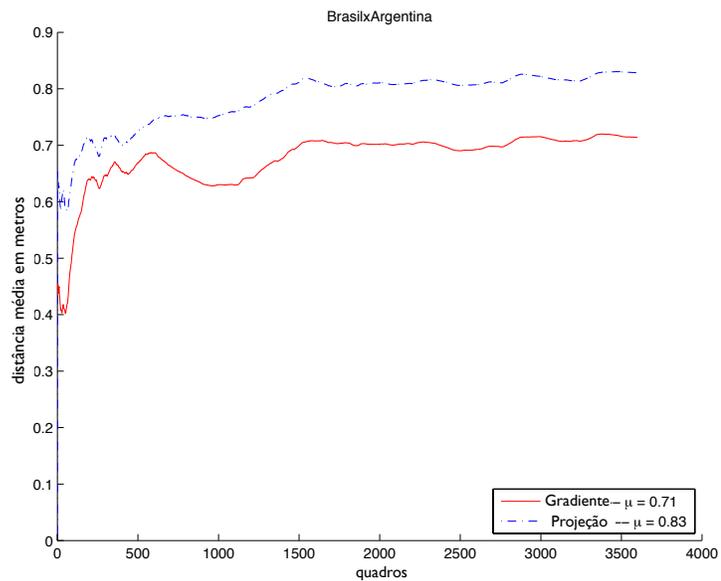


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 1, tempo 1 do jogo BrasilArgentina.

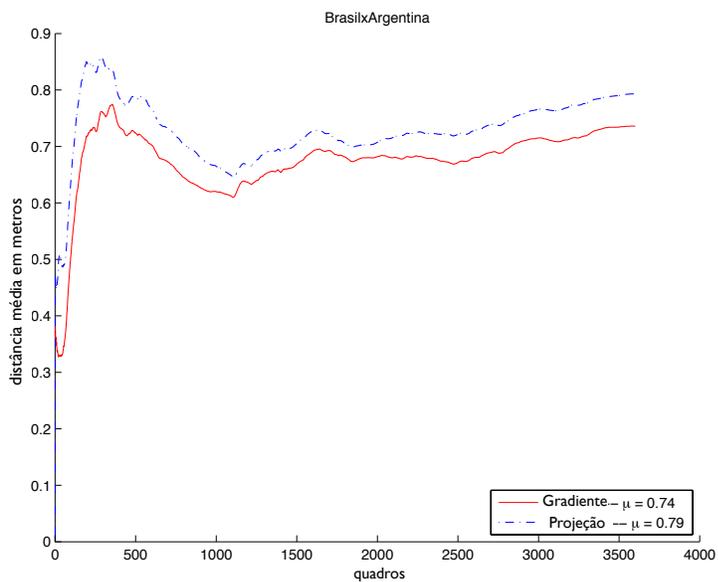


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 2, tempo 1 do jogo BrasilxArgentina.

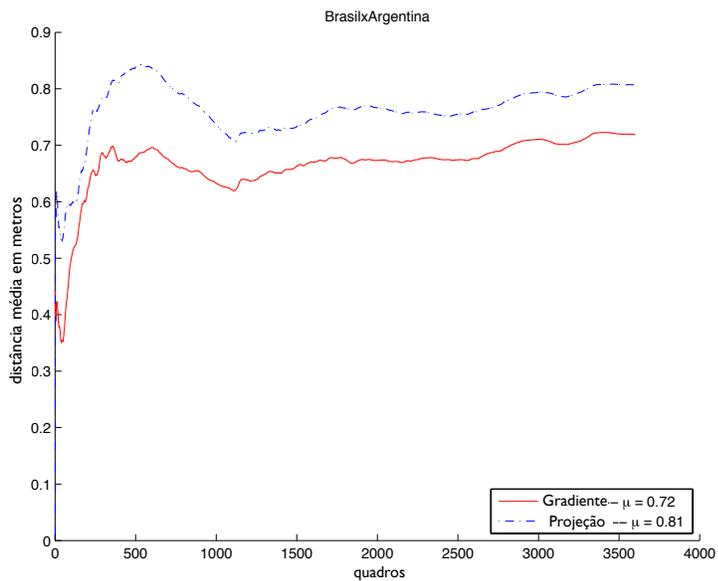


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 3, tempo 1 do jogo BrasilxArgentina.

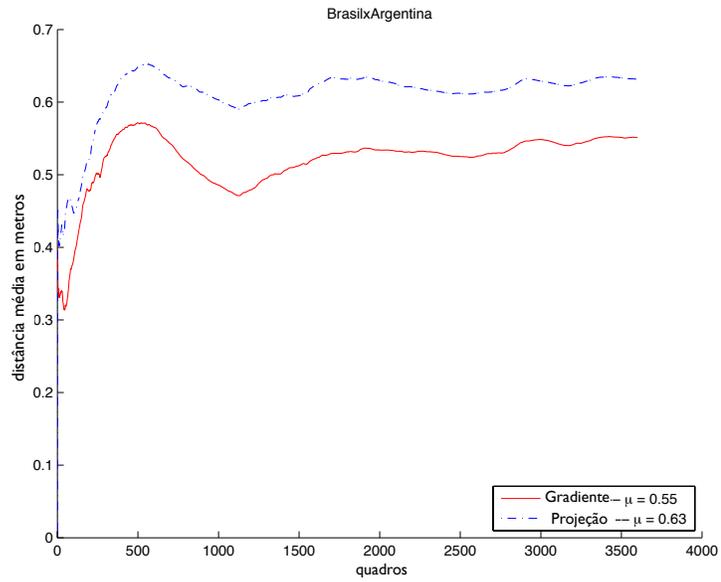


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 4, tempo 1 do jogo BrasilxArgentina.

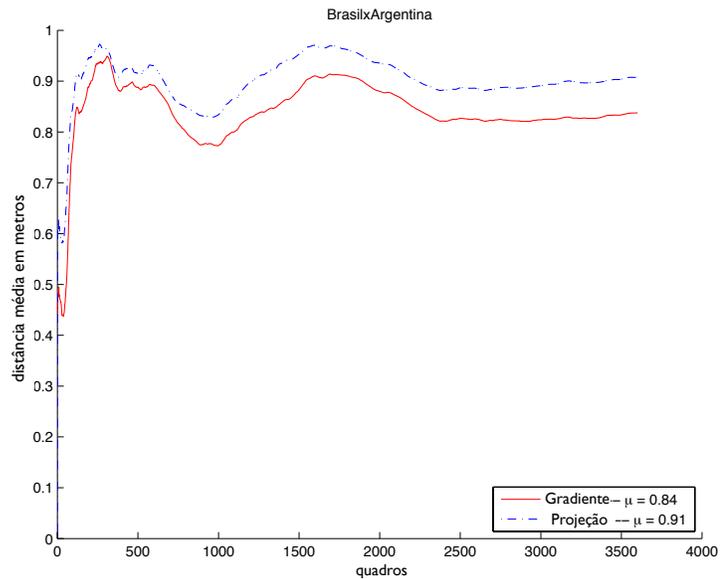


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 1, tempo 2 do jogo BrasilxArgentina.

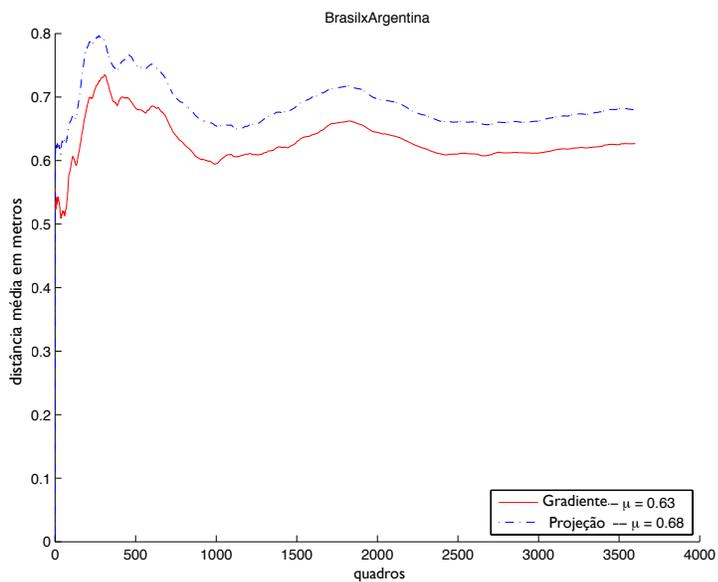


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 2, tempo 2 do jogo BrasilxArgentina.

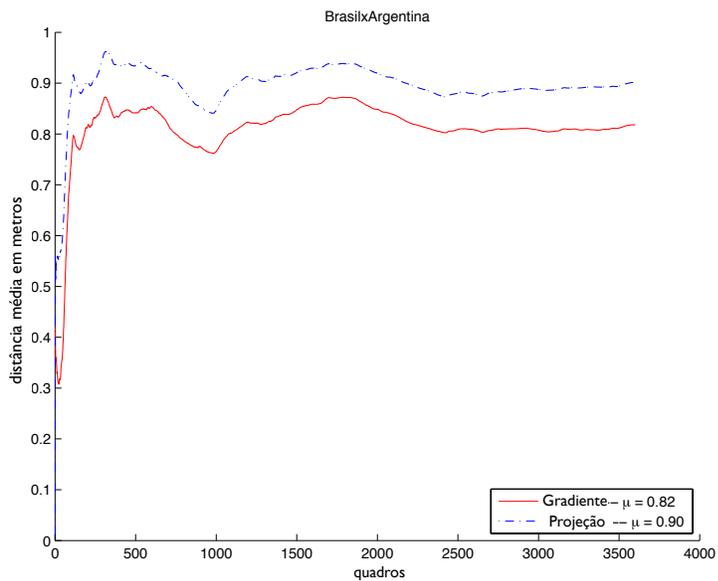


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 3, tempo 2 do jogo BrasilxArgentina.

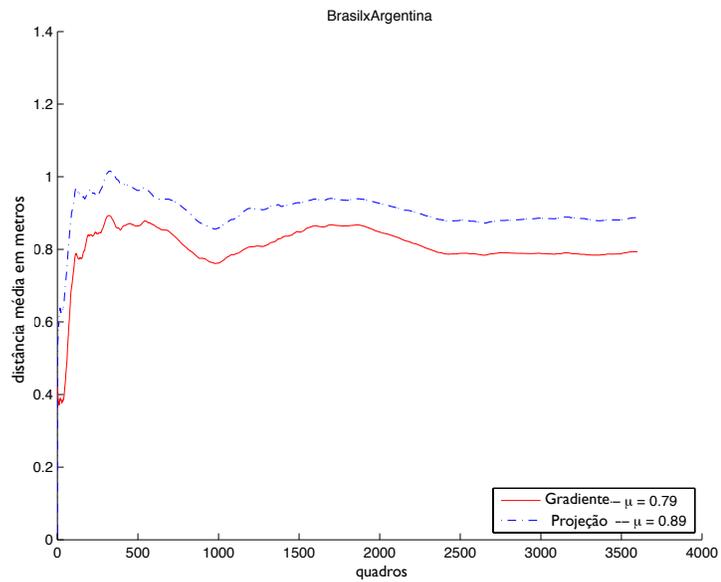


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 4, tempo 2 do jogo BrasilxArgentina.

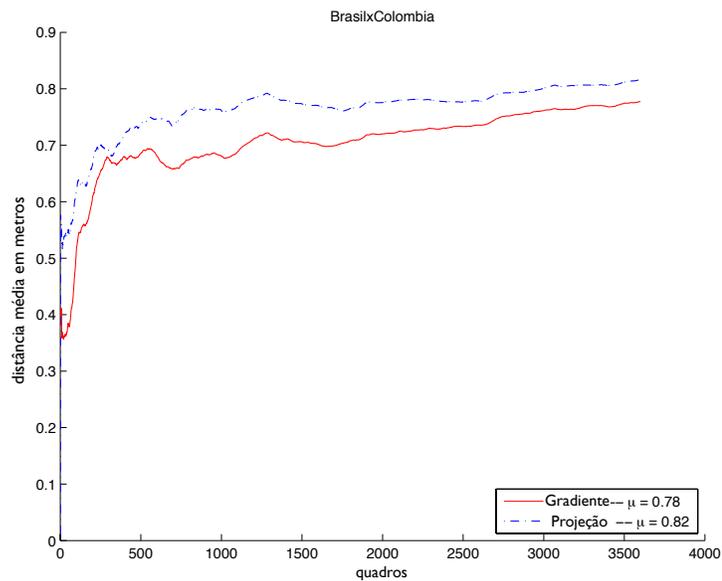


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 1, tempo 1 do jogo BrasilxColombia.

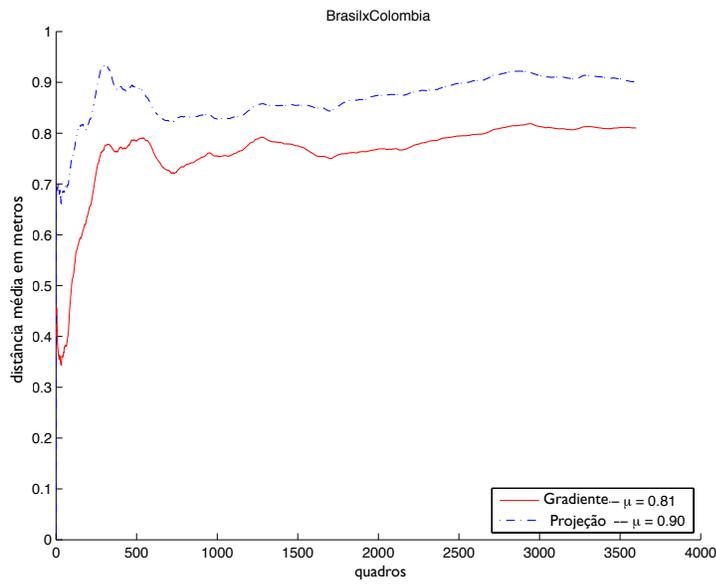


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 2, tempo 1 do jogo BrasilxColombia.

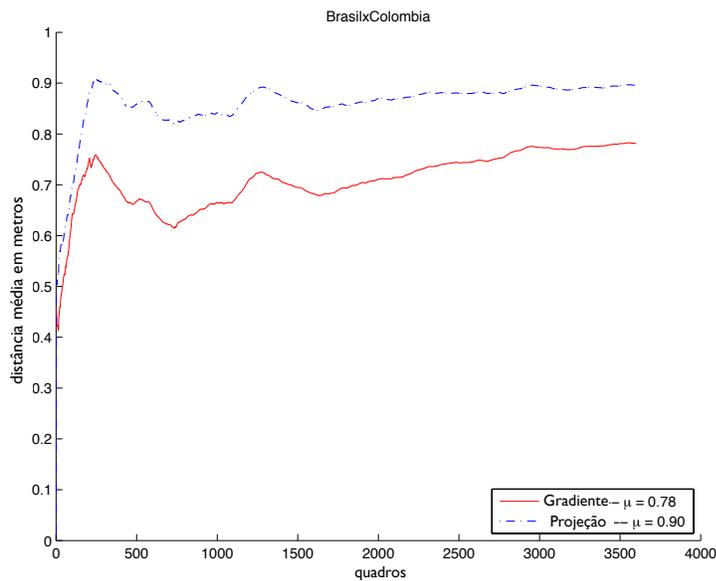


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 3, tempo 1 do jogo BrasilxColombia.

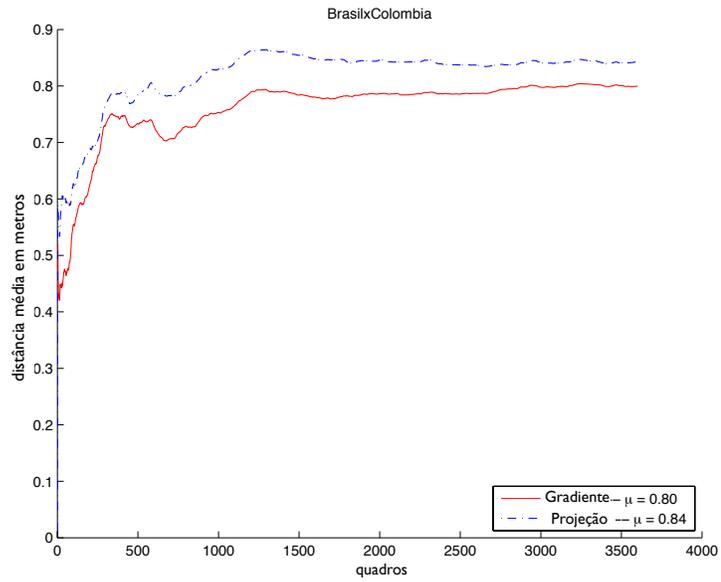


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 4, tempo 1 do jogo BrasilxColombia.

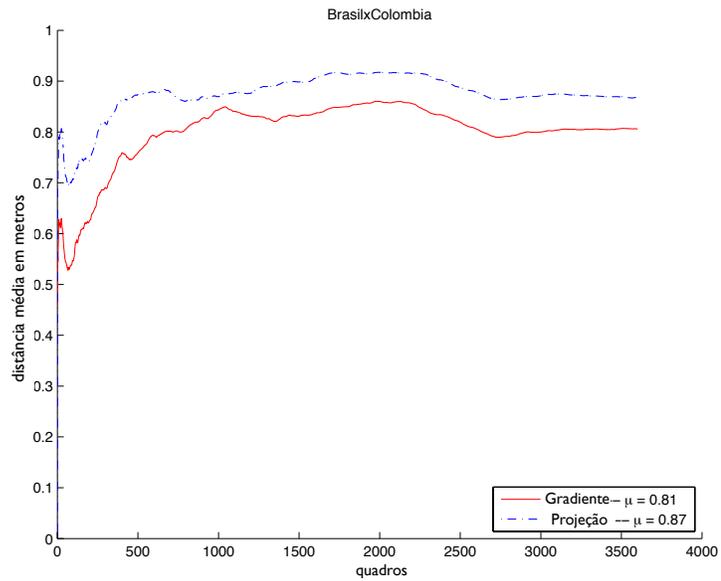


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 1, tempo 2 do jogo BrasilxColombia.

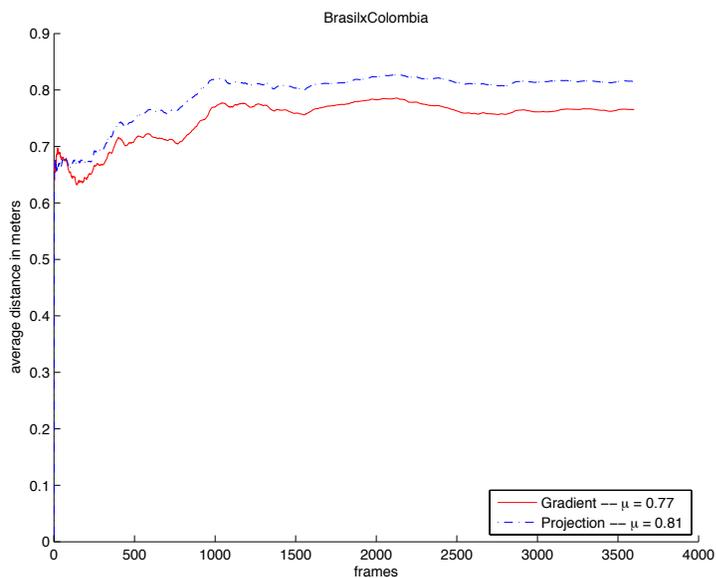


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 2, tempo 2 do jogo BrasilxColombia.

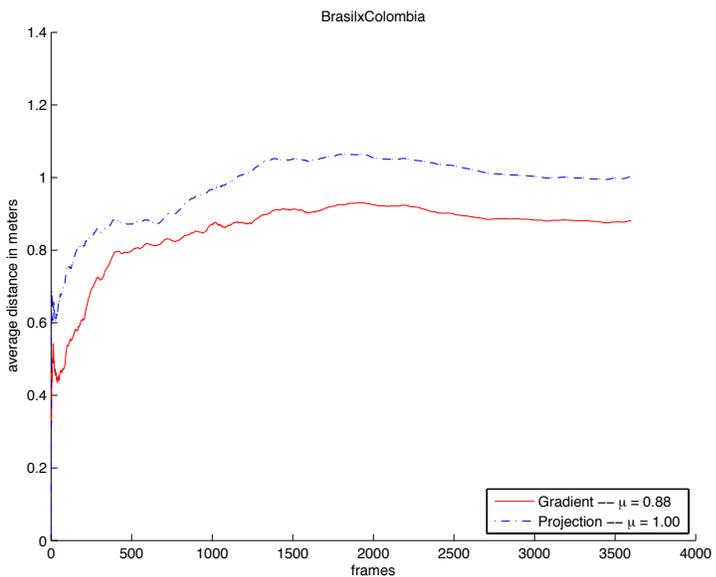


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 3, tempo 2 do jogo BrasilxColombia.

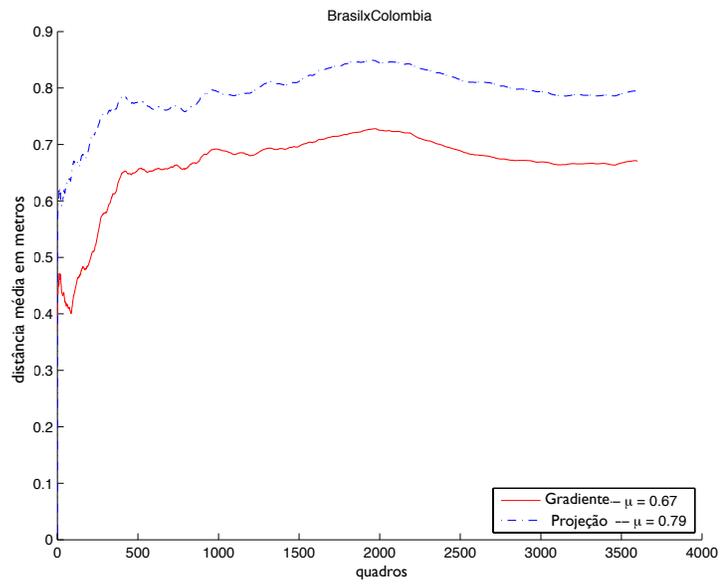


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 4, tempo 2 do jogo BrasilxColombia.

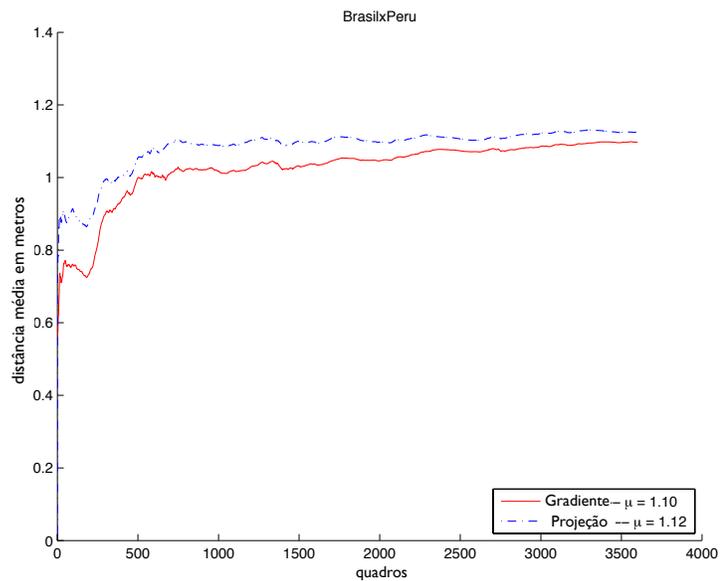


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 1, tempo 1 do jogo BrasilxPeru.

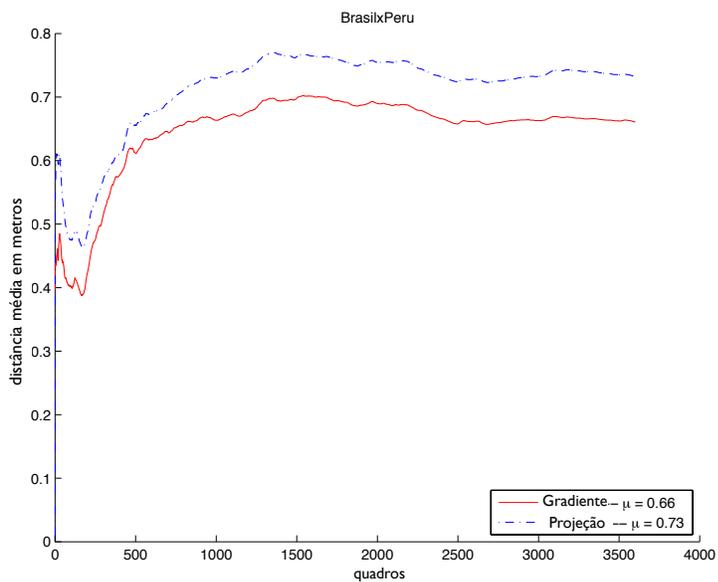


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 2, tempo 1 do jogo BrasilxPeru.

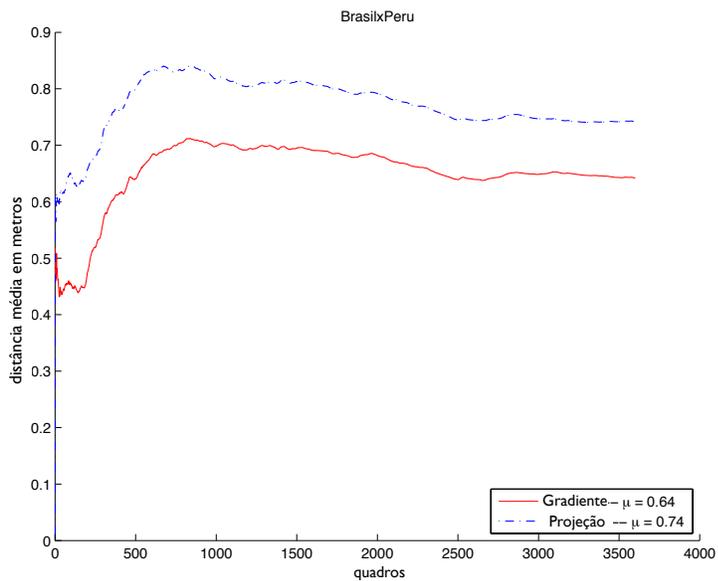


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 3, tempo 1 do jogo BrasilxPeru.

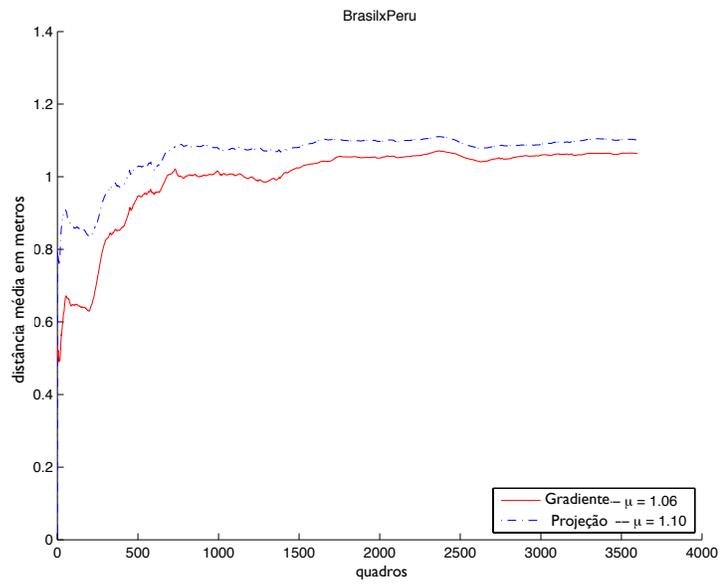


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 4, tempo 1 do jogo BrasilxPeru.

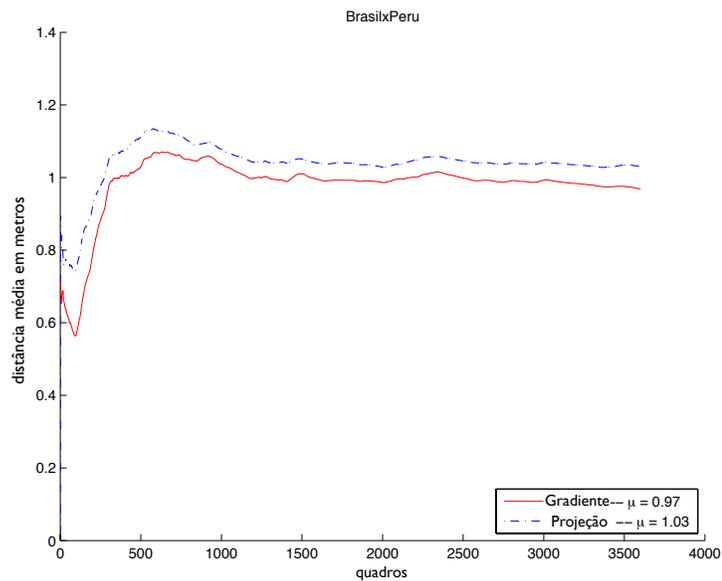


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 1, tempo 2 do jogo BrasilxPeru.

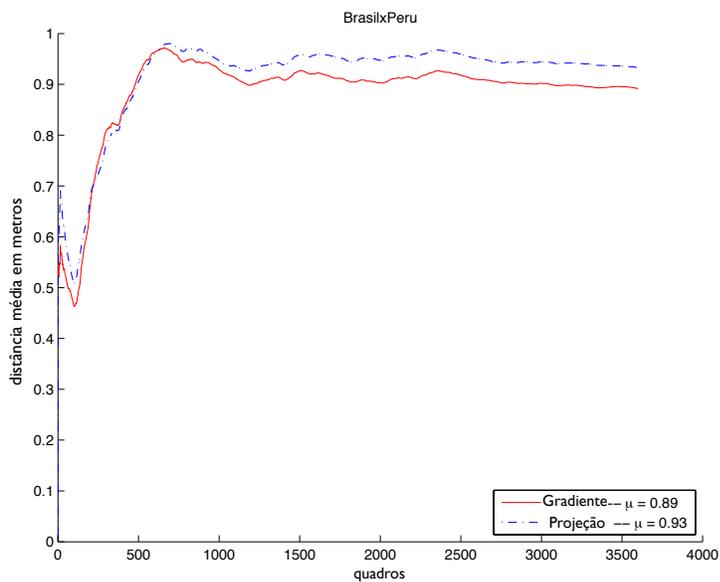


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 2, tempo 2 do jogo BrasilxPeru.

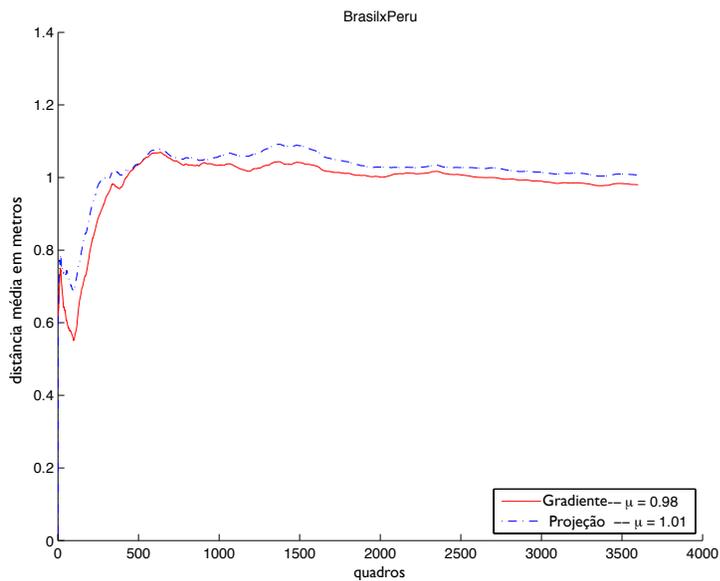


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 3, tempo 2 do jogo BrasilxPeru.

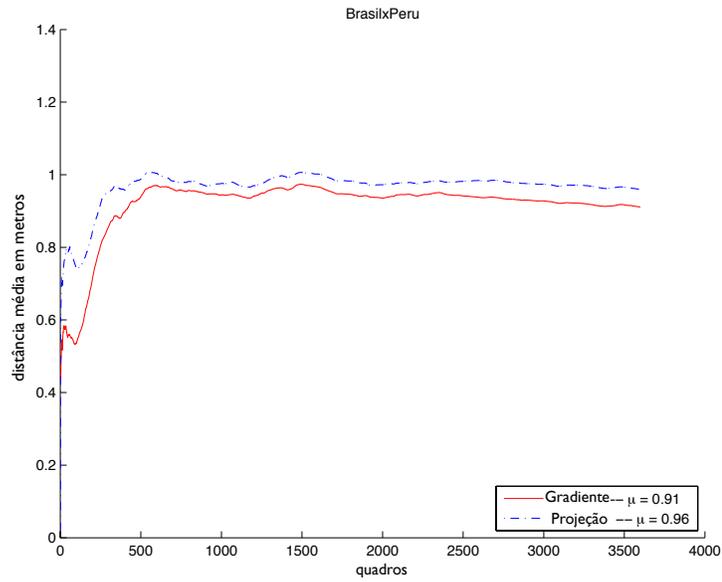


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 4, tempo 2 do jogo BrasilxPeru.

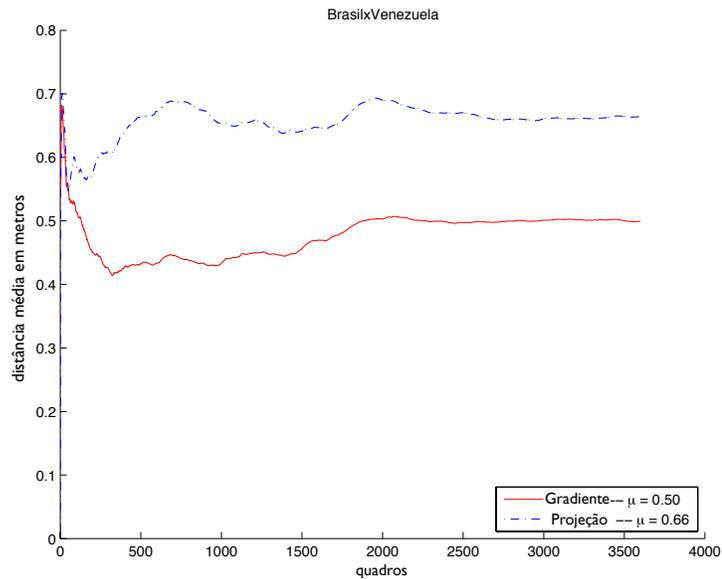


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 1, tempo 1 do jogo BrasilxVenezuela.

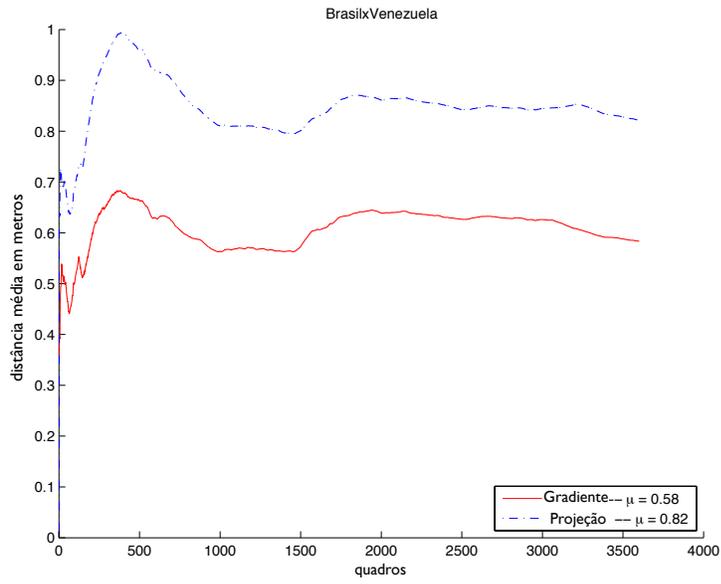


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 2, tempo 1 do jogo BrasilxVenezuela.

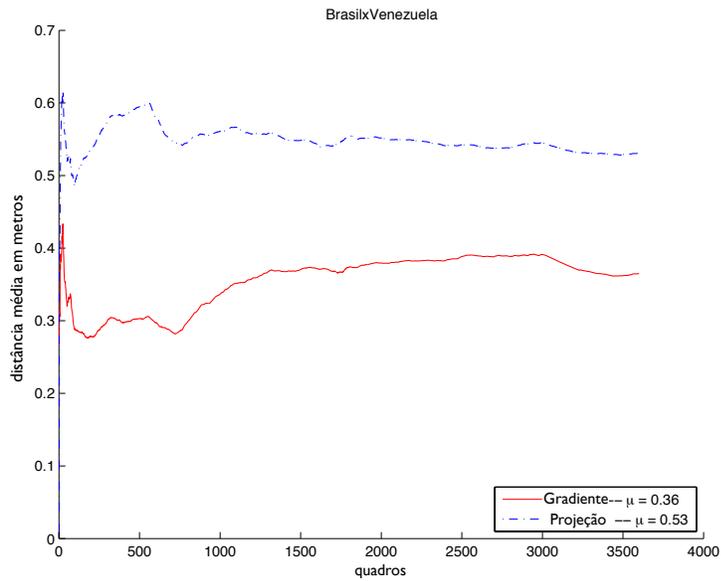


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 3, tempo 1 do jogo BrasilxVenezuela.

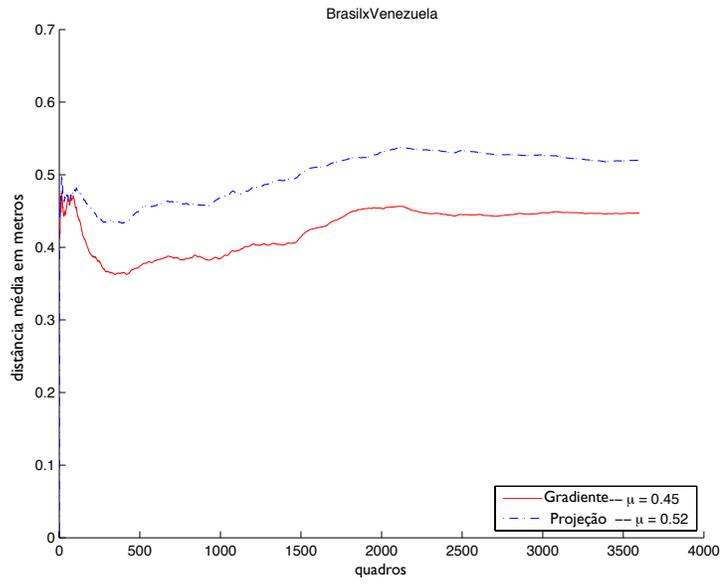


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 4, tempo 1 do jogo BrasilxVenezuela.

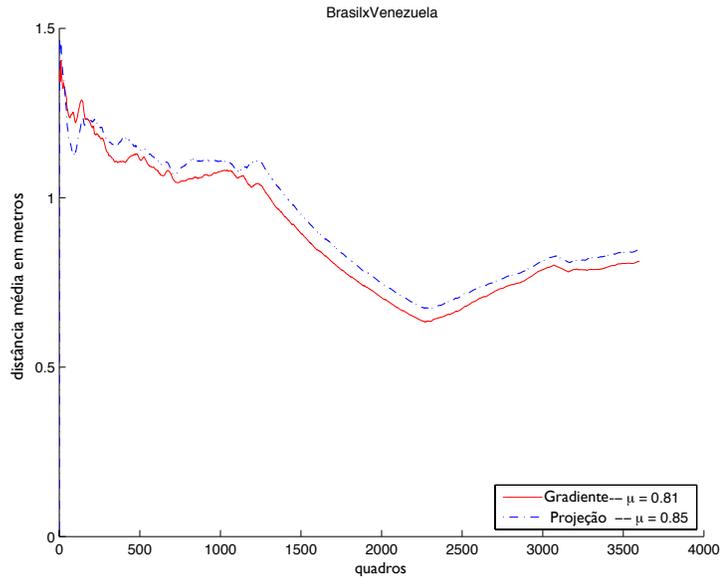


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 1, tempo 2 do jogo BrasilxVenezuela.

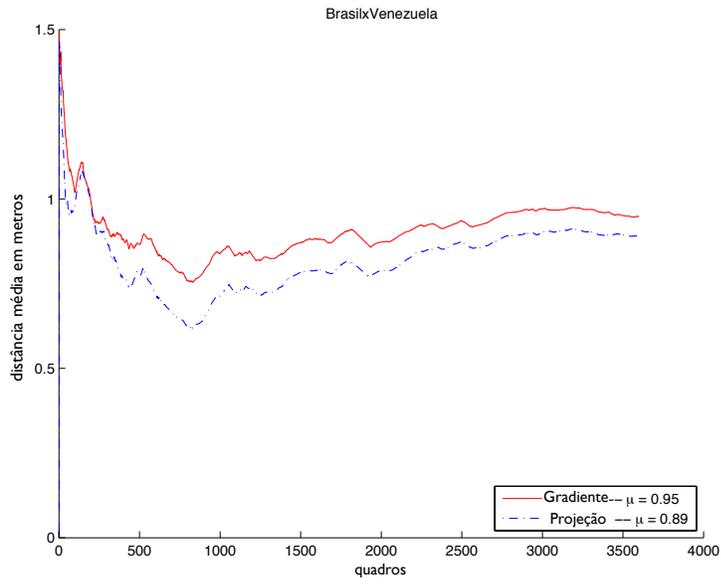


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 2, tempo 2 do jogo BrasilxVenezuela.

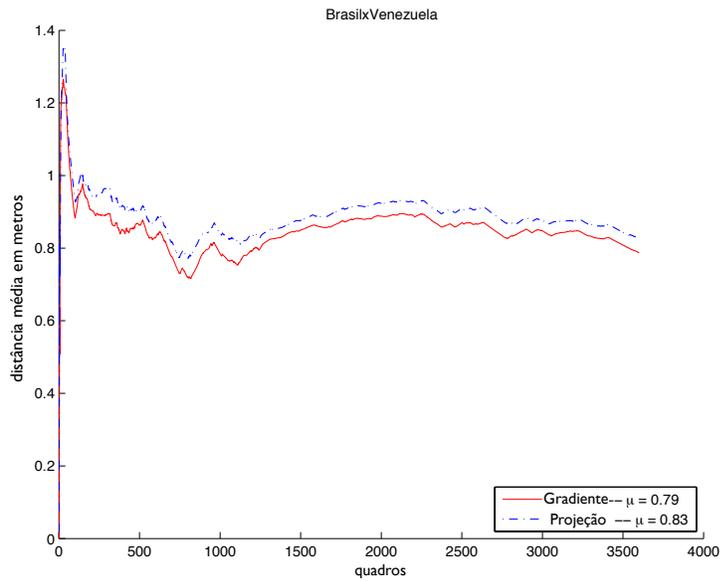


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 3, tempo 2 do jogo BrasilxVenezuela.

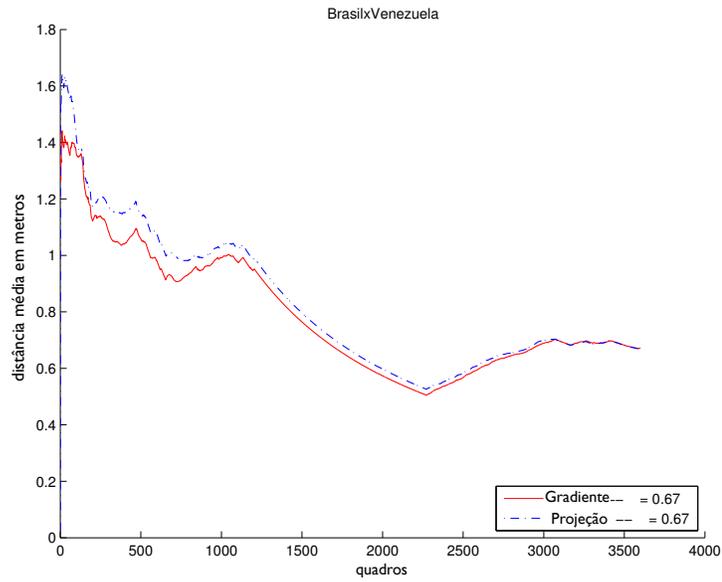


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 4, tempo 2 do jogo BrasilxVenezuela.

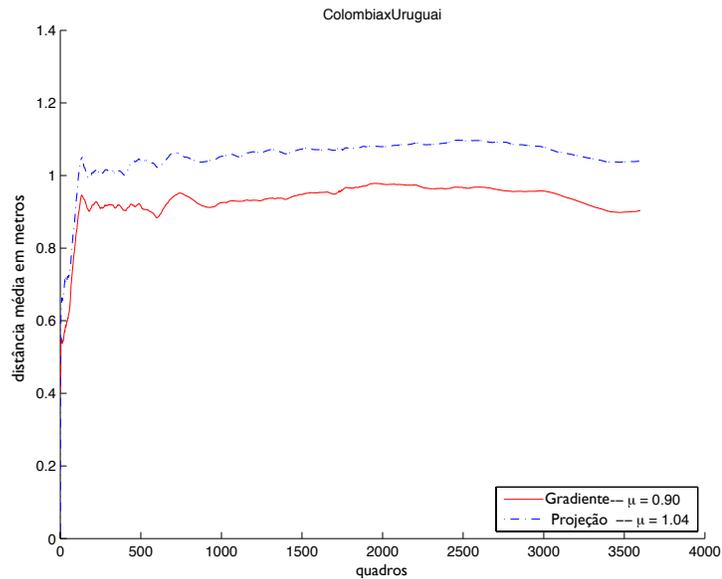


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 1, tempo 1 do jogo ColombiaxUruguai.

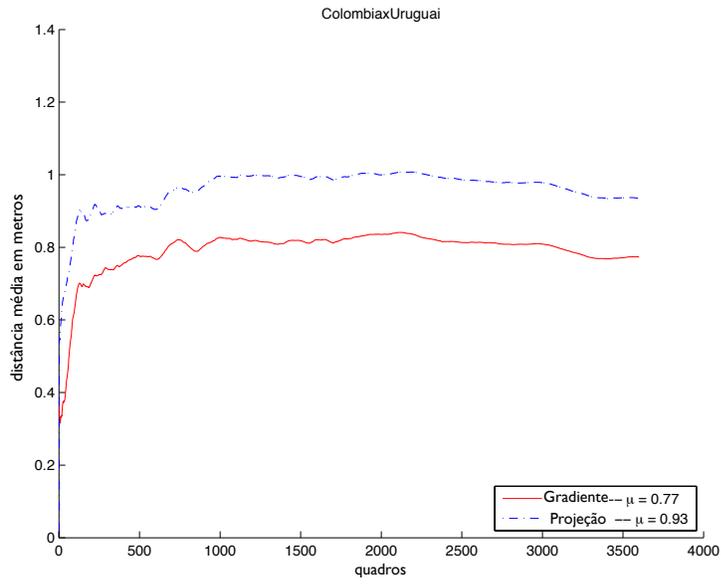


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 2, tempo 1 do jogo ColombiáxUruguai.

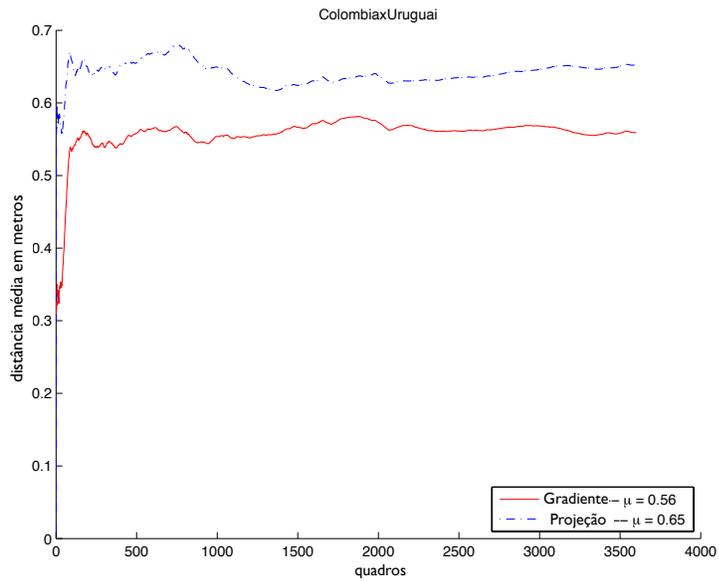


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 3, tempo 1 do jogo ColombiáxUruguai.

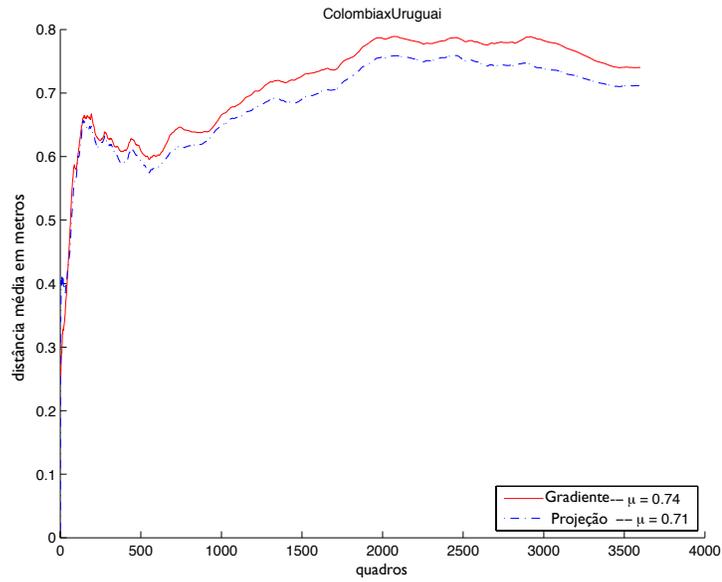


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 4, tempo 1 do jogo ColombiaxUruguai.

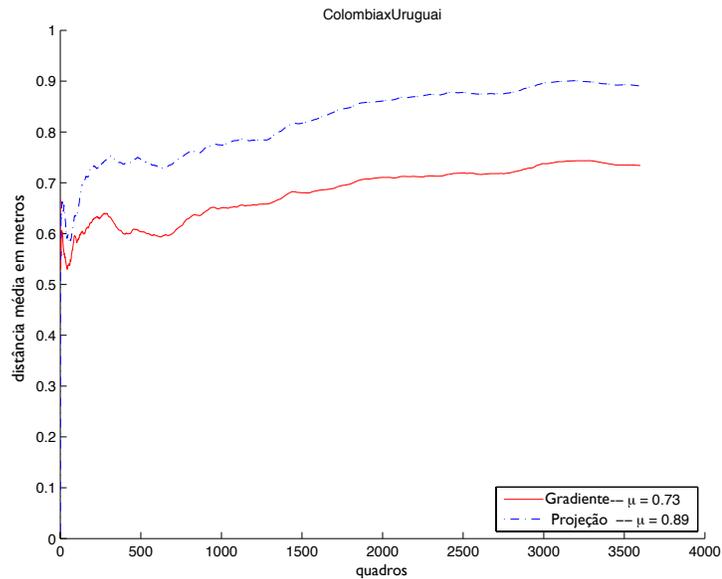


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 1, tempo 2 do jogo ColombiaxUruguai.

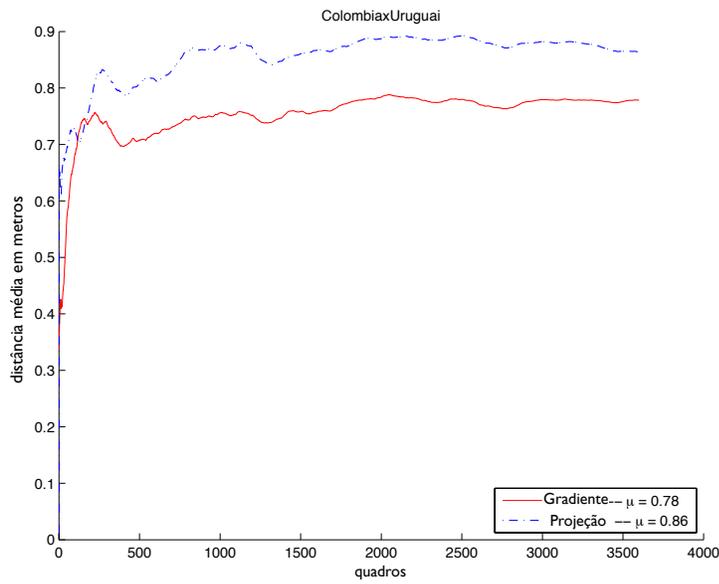


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 2, tempo 2 do jogo ColombiáxUruguai.

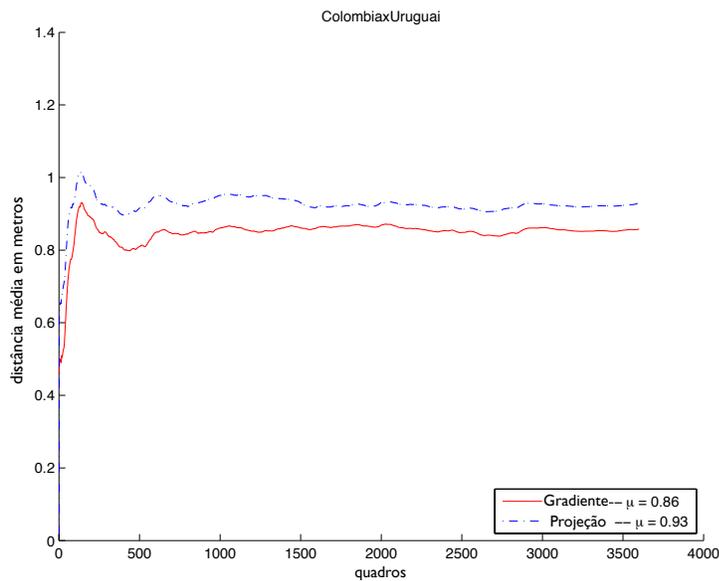


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 3, tempo 2 do jogo ColombiáxUruguai.

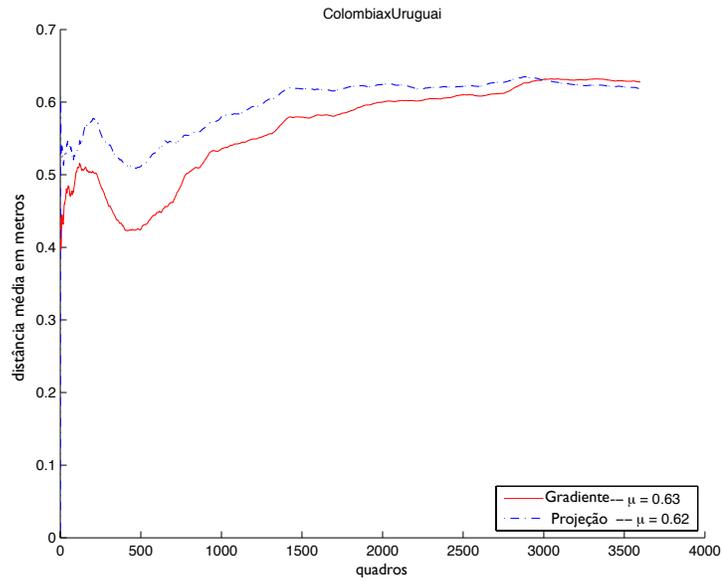


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 4, tempo 2 do jogo ColombiaxUruguai.

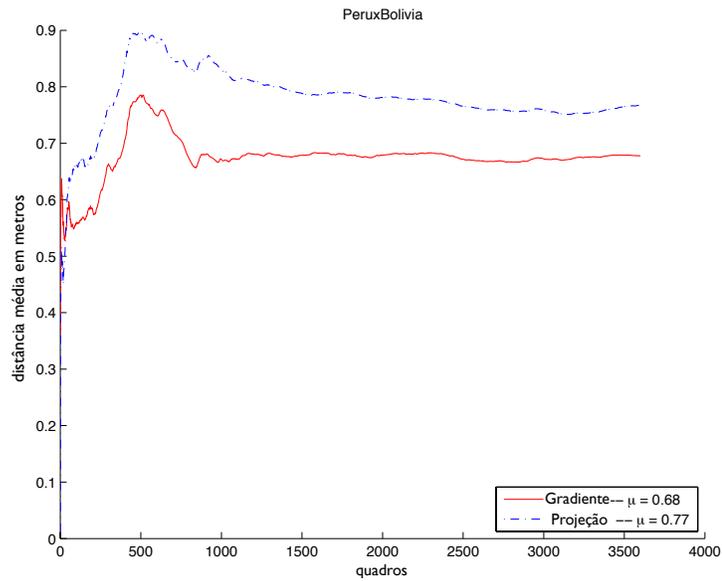


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 1, tempo 1 do jogo PeruxBolivia.

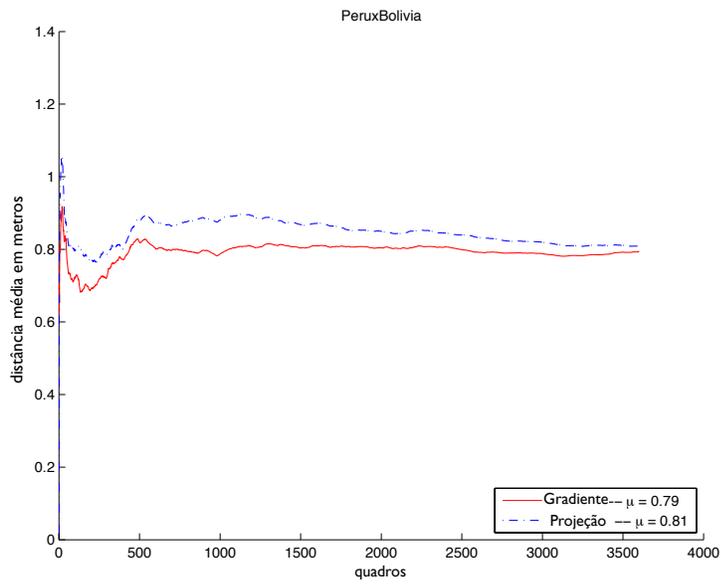


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 2, tempo 1 do jogo PeruxBolivia.

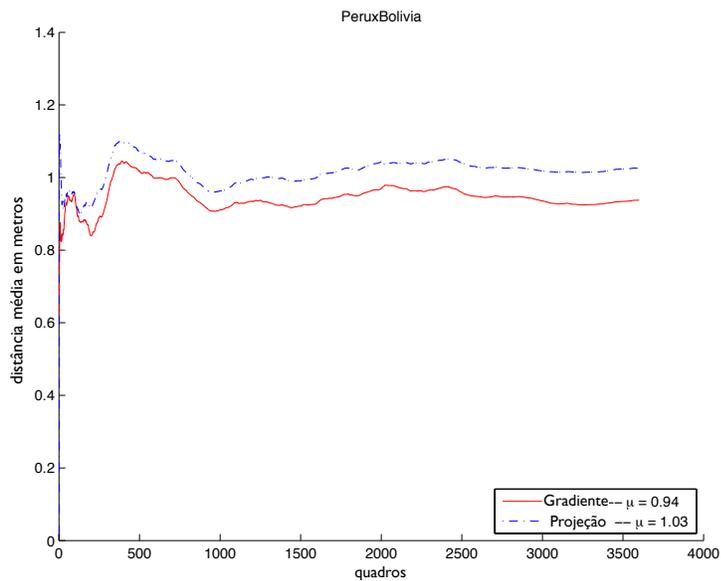


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 3, tempo 1 do jogo PeruxBolivia.

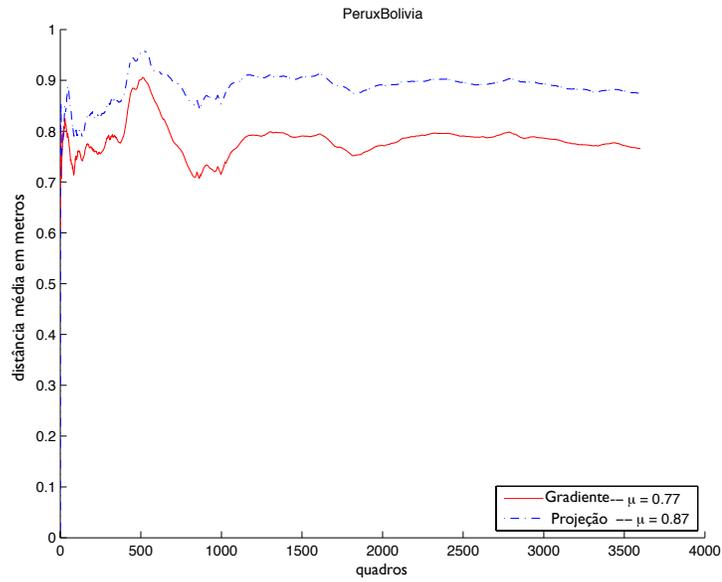


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 4, tempo 1 do jogo PeruxBolívia.

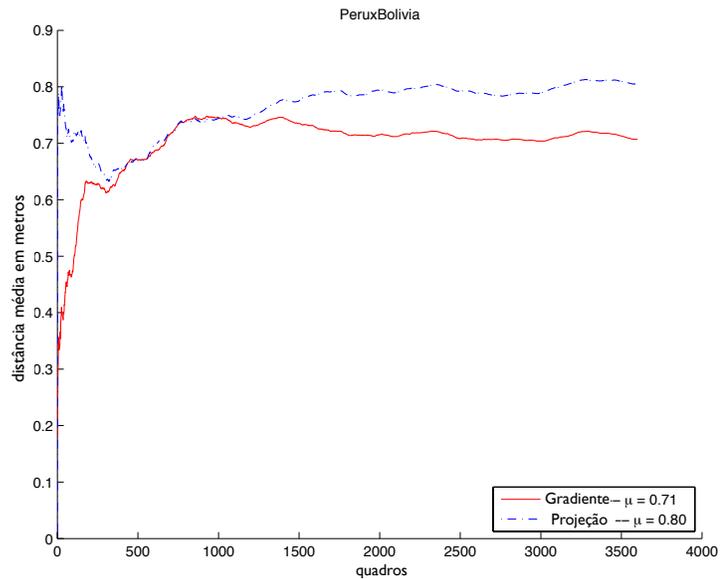


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 1, tempo 2 do jogo PeruxBolívia.

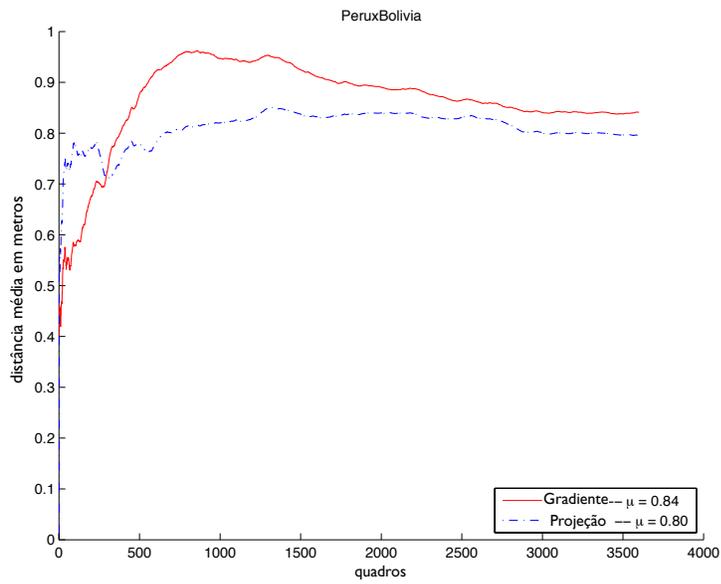


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 2, tempo 2 do jogo PeruxBolivia.

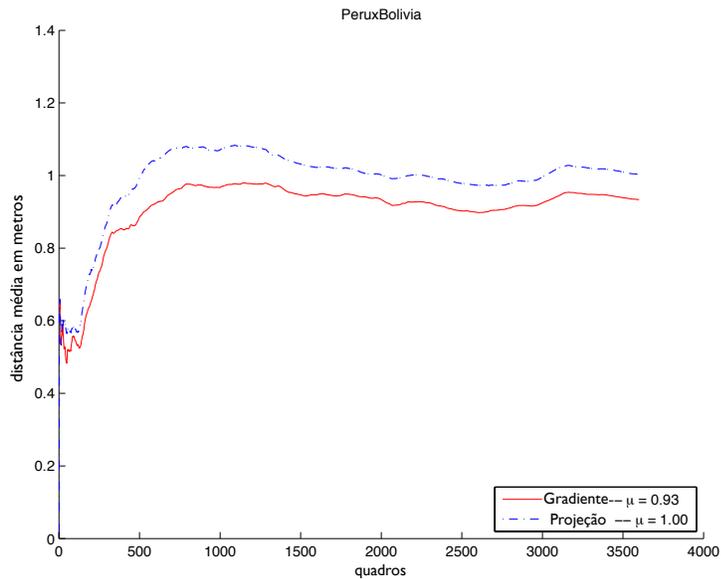


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 3, tempo 2 do jogo PeruxBolivia.

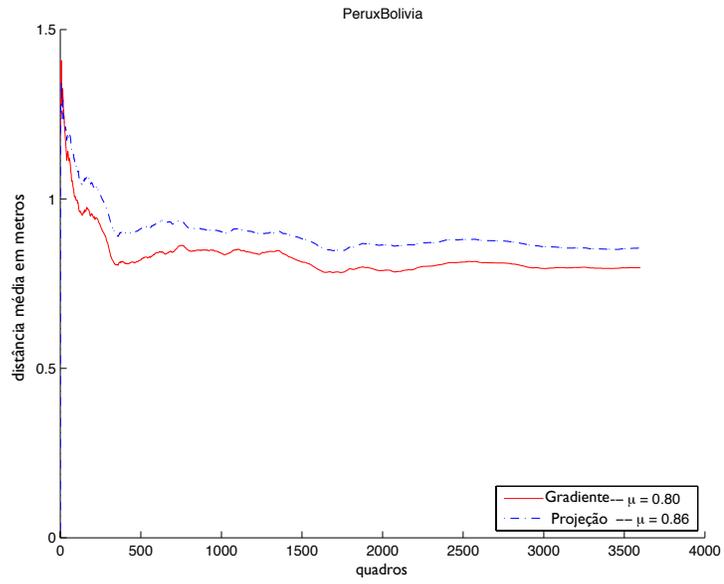


Figura A.1 (Continuação): Erro médio na estimativa observado na câmera 4, tempo 2 do jogo PeruxBolívia.