

Modeling of Dynamic Environments for Visual Forecasting of American Football Plays

Namhoon Lee

CMU-CS-15-140

December 2015

Computer Science Department
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

Thesis Committee:

Kris M. Kitani, Chair

Martial Hebert

Sebastian Scherer

*Submitted in partial fulfillment of the requirements
for the degree of Master of Science*

Copyright © 2015 Namhoon Lee

This research was supported in part by the Japanese Science and Technology (JST) CREST grant and the Graduate Study Abroad Scholarship from Jeongsong Cultural Foundation, South Korea.

The views and conclusions contained in this document are those of the author and should not be interpreted as representing the official policies, either expressed or implied, of any sponsoring institution, the U.S. government, or any other entity.

Keywords: Human Activity Forecasting, American Football, Dynamic Environment, Dynamic Feature, Inverse Optimal Control, Sequential Inference

To my family and friends, for their endless support and encouragement.

Abstract

Predicting the trajectory of a wide receiver in the game of American football requires prior knowledge about the game (*e.g.*, route trees, defensive formations) and an accurate model of how the environment will change over time (*e.g.*, opponent reaction strategies, motion attributes of players). Our aim is to build a computational model of the wide receiver, which takes into account prior knowledge about the game and short-term predictive models of how the environment will change over time. While prior knowledge of the game is readily accessible, it is quite challenging to build predictive models of how the environment will change over time. We propose several models for predicting short-term motions of opponent players to generate dynamic input features for our wide receiver forecasting model. In particular, we model the wide receiver with a Markov Decision Process (MDP), where the reward function is a linear combination of static features (prior knowledge about the game) and dynamic features (short-term prediction of opponent players). Since the dynamic features change over time, we make recursive calls to an inference procedure over the MDP while updating the dynamic features. We validate our technique on a video dataset of American football plays. Our results show that more informed models that accurately predict the motions of the defensive players are better at forecasting wide receiver plays.

Acknowledgments

First, I would like to thank my advisor, Kris M. Kitani, for his countless support and encouragement. He enriched my research with his insightful advice and taught me how to find and solve important problems. His patience and generosity enabled my persistence to perform rigorous research and made me never give up even when things were rough. My research could have never been lucrative without his support. I would also like to thank Martial Hebert and Sebastian Scherer for being my thesis committee on a short request and sharing valuable comments and suggestions on this work.

In addition, I would like to thank all my lab mates, De-An Huang, Akanksha Saran, Minghuang Ma, Wei-Chiu Ma, Rohit Girdhar, Ankit Laddha, Hanbyul Joo, Wen-Sheng Chu, Minh Vo, Yuxiong Wang and all the people in the Computer Vision Group at CMU for being always kind to me and sharing valuable ideas and discussions.

Furthermore, I would like to thank Changho Oh, June Young Park, Yohan Jo, and all my Korean friends for true friendship. Since the beginning of my study at CMU, they made my graduate life multifarious. I would not have been able to sustain the rough life at CMU without the joy of being with them.

Lastly, but not least, I am thankful to CMU for all the invaluable opportunities given to me. I thank David A. Eckhardt for his guidance on my progress and Tracy Farbacher for her kindness whenever I need a help. Thanks to their help, I was able to enjoy my journey at CMU and build a successful academic career.

Contents

- 1 Introduction** **1**
 - 1.1 Overview 1
 - 1.2 Background: American Football 3
 - 1.3 Static Environments VS. Dynamic Environments 3
 - 1.4 Related Work 4
 - 1.5 Contributions 5
 - 1.6 Road Map 6

- 2 Modeling of Forecasting Environments** **7**
 - 2.1 Modeling of Player Activity 7
 - 2.2 Modeling of Static Environments 8
 - 2.3 Prediction of Dynamic Environments 9
 - 2.3.1 Non-linear feature regressor 11
 - 2.3.2 Linear (constant motion) feature regressor 13

- 3 Forecasting in Dynamic Environments** **15**
 - 3.1 Maximum entropy inverse optimal control 15
 - 3.2 Sequential inference in dynamic environment 16
 - 3.3 Algorithm 17

- 4 Experimental Results** **21**
 - 4.1 Setup 21
 - 4.1.1 Dataset 21
 - 4.1.2 Feature maps 22
 - 4.1.3 Metrics and measurements 22
 - 4.2 Evaluation of short-term dynamics predictors 24
 - 4.3 Evaluation of activity forecasting 27
 - 4.4 Application to Sports Analytics 28

- 5 Conclusion** **31**

- Bibliography** **33**

List of Figures

1.1	An illustration of human activity forecasting with the case of a wide receiver in American football. The goal of this work is to infer the agent’s activity from an image: “Which path will the wide receiver choose?”	2
1.2	Our approach forecasts viable trajectories for the wide receiver in American football.	2
1.3	Static Environment VS Dynamic Environment. In static environment, the wide receiver (WR, cyan dot) will proceed straight to the goal (★) since defender blocking in his path is set immovable. In dynamic environment, however, the wide receiver makes a detour to avoid collision with the cornerback (CB, magenta triangle).	4
2.1	A representation of Markov decision process.	8
2.2	A state representation by a grid map over the football field.	9
2.3	Reinforcement learning: learning takes place as a result of interaction between an agent and the environment.	10
2.4	A basic route tree.	10
2.5	A posteriori trajectory pattern of wide receiver.	11
2.6	Creation of a training example used in non-linear feature regressor. Both wide receiver (cyan) and cornerback (magenta) are contained in \mathbf{x} implicitly to predict the opponent reaction.	12
4.1	Registration of an American football video onto a top-down view ground field. Trajectory of the wide receiver (cyan) and the cornerback (magenta).	22
4.2	(a)-(e) Five types of feature response maps for an example scene. Each pixel of the map indicates a possible state s (represented as $[x, y]$ location in the map) that people can reach to and has a corresponding feature response value $\mathcal{F}(s)$. (f) A reward map computed using feature response maps with corresponding reward weights learned in training phase. Red region indicates the states with high reward, and the offense proceeds to the right in this scene.	23
4.3	Average prediction errors of static and dynamic features. The prediction errors in dynamic environment are computed by the sequential inference procedures, while the error in static environment is computed by setting the initial location of cornerback fixed throughout the play.	25
4.4	(a-c) Average errors of GPR regressor in varying parameters. (d) The distance to goal for the WR and CB.	26

4.5	Selected forecasting results in comparison. The forecast distributions using the proposed CM (b) and the proposed GPR (c) match much closer to the ground-truth wide receiver trajectories (red) than the regular forecasting approach (a). Best viewed in electronic form.	27
4.6	Average KLD over all test scenes from destination forecasting. The error decreases as play comes to the end.	29
4.7	A destination forecasting result. The forecast distribution becomes closer to the demonstrated trajectory (red) as more observations are revealed by time. Best viewed in electronic form.	30

List of Tables

4.1 Average errors of forecasting approaches measured in different metrics. The proposed methods, CM and GPR, outperform the standard forecasting model [17] in all criteria. 28

Chapter 1

Introduction

1.1 Overview

The task of analyzing human activity has received much attention in the field of computer vision. Among the many sub-fields dealing with human activities, we address the problem of *activity forecasting* which refers to the task of inferring the future actions of people from visual input [17]. Human activity forecasting is a different task from that of recognition, detection or tracking. Vision-based activity forecasting aims to predict how an agent will act in the future given a single image of the world (see Figure 1.1).

Forecasting human activity in *dynamic* environments is extremely difficult because changes in the environment must also be hallucinated. Recent work on forecasting activity [17] has been limited to *static* environments where the reward function defined over the state space does not change (*i.e.*, objects in the scene are assumed to be immovable), making it easier to reason about future action sequences. Predicting the future becomes difficult if the environment keeps changing over time, since the agent will have to take into account the possible repercussions of his actions on others. This is especially true in sports scenarios: an offensive play will evolve differently depending on the response of the defense over time. In this work, we focus on forecasting human activity in the dynamic domain of sports by iteratively predicting short-term changes in the environment.

To estimate future changes in the environment (the defense), we utilize two different short-term dynamics prediction models: (1) a non-linear model exploiting Gaussian process regression (GPR) and (2) a linear model based on a constant motion constraint on human motion (CM).

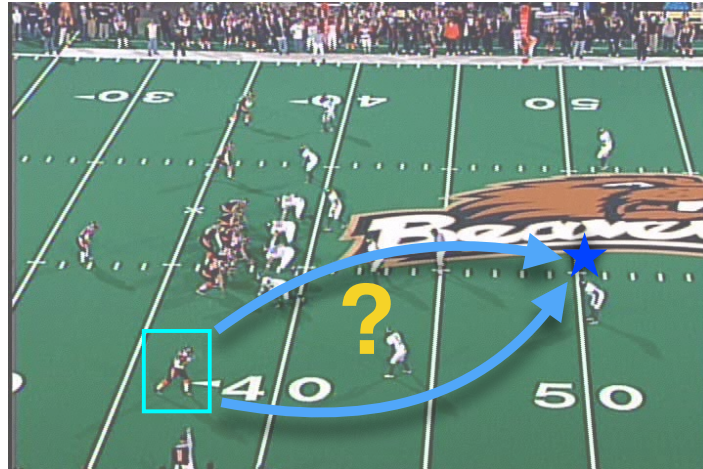


Figure 1.1: An illustration of human activity forecasting with the case of a wide receiver in American football. The goal of this work is to infer the agent’s activity from an image: “Which path will the wide receiver choose?”

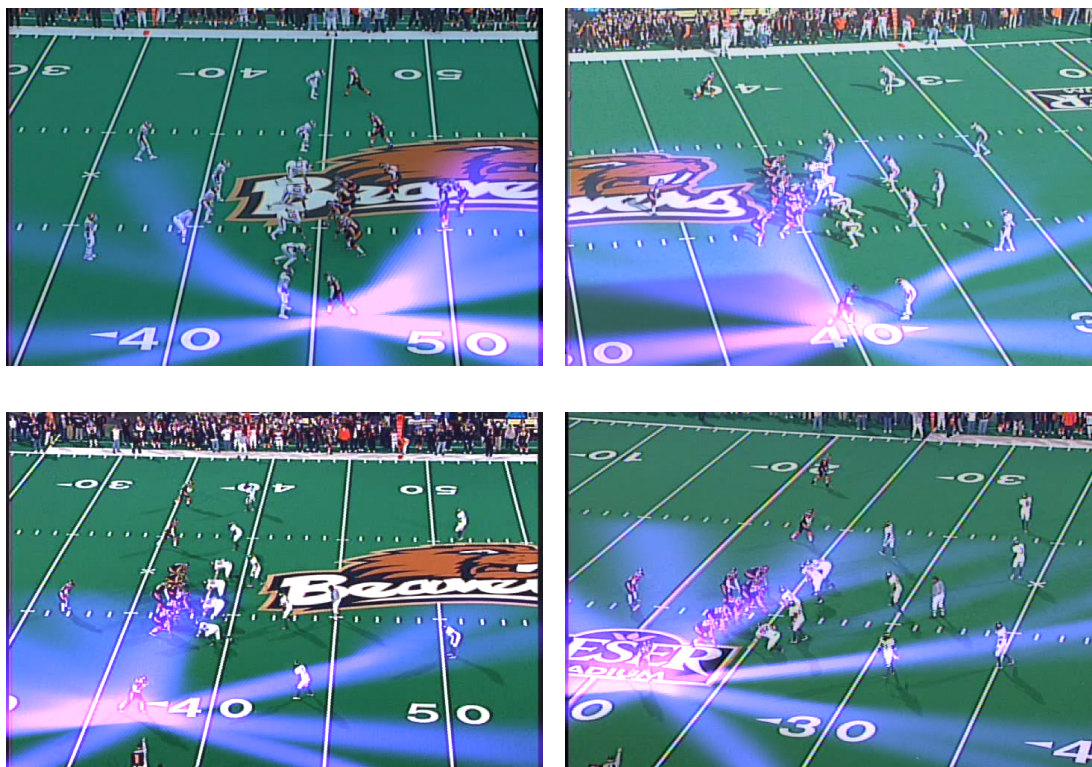


Figure 1.2: Our approach forecasts viable trajectories for the wide receiver in American football.

To process the output of the short-term dynamics predictors, we develop a sequential inference algorithm that incrementally forecasts future actions of the wide receiver based on the updated set of short-term predictions of defensive motion. We stress here that we only hallucinate defensive motion and we do not use true observations of the defense (the proposed algorithm does not have access to the future). Our approach enables forecasting in dynamic environment and outperforms the standard state-of-the-art forecasting model in predicting wide receiver activity in terms of both qualitative and quantitative measurements. To the best of our knowledge, this is the first work to predict wide receiver trajectories from images, by taking into account the incremental dynamics of defensive play.

1.2 Background: American Football

In American football, there are offensive and defensive teams of 11 players each. The offense attempts to advance toward the end zone by running with or passing the ball while the team without control of the ball, the defense, aims to stop their advance and take control of the ball for themselves [26].

The wide receiver is an offensive position that plays a key role in *passing plays*. In order to succeed in passing plays, the wide receiver attempts to receive the ball from the quarterback, while avoiding or outrunning defenders (typically cornerbacks or safeties) along his pass route. A cornerback is a defensive role that covers the wide receiver and attempts to block any passing plays. It is vital for the wide receiver to avoid the defense and advance as far as possible. Since the movement of the wide receiver can be different depending on the defenders, predicting the wide receiver trajectories should take into account the possible changes to the environment (see Figure 1.3).

1.3 Static Environments VS. Dynamic Environments

To simplify our problem, we build our proposed prediction model based on an assumption that the cornerback is the primary defender affecting the wide receiver's future activity and contributes to the changes in the environment (*i.e.*, generates dynamic features). In our scenario, the opponent (CB) forms a negative force field which repells the wide receiver.

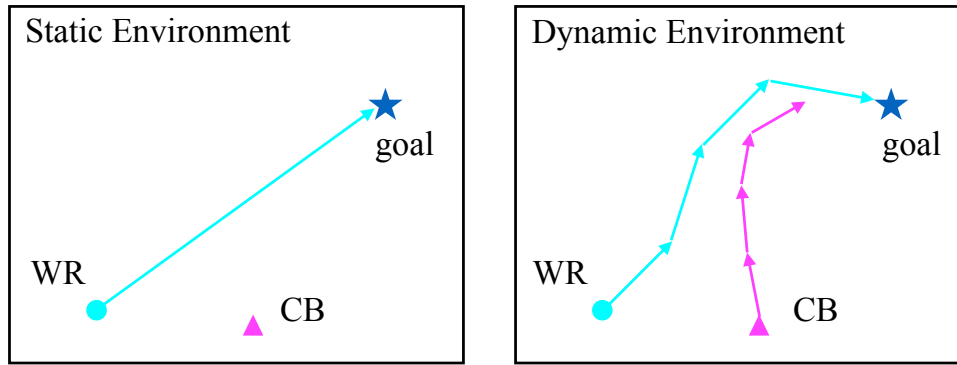


Figure 1.3: Static Environment VS Dynamic Environment. In static environment, the wide receiver (WR, cyan dot) will proceed straight to the goal (★) since defender blocking in his path is set immovable. In dynamic environment, however, the wide receiver makes a detour to avoid collision with the cornerback (CB, magenta triangle).

1.4 Related Work

Human activities in sports scenes have been analyzed through a variety of computer vision techniques for different purposes. Intille and Bobick [10] proposed a recognition algorithm for identifying team-based action in the domain of American football. [20, 21] also modeled opponent players of American football successfully, but they used real-time responses to learn policies in an artificial game setting. Previous work [13, 15] used motion fields to predict play evolution in dynamic sports scenes and predicted global centers of focus. Lucey *et al.* [23] effectively discovered team formation and plays using the role representation. For the purpose of predicting agent motion, not necessarily in sports, Kim *et al.* [16] predicted agent motion using a Kalman Filter based on reciprocal velocity obstacles. Pellegrini *et al.* [27] used the stochastic linear trajectory avoidance model to predict pedestrian trajectories. Urtasun *et al.* [34] proposed a form of Gaussian process dynamical models for learning human motion, and the authors in [14] used Gaussian process regression to learn motion trajectories, classify the pattern and detect anomalous events. It is worth noting that we predict wide receiver trajectories from an image, and Walker *et al.* [36] also proposed visual prediction from images using mid-level visual patches. In [12, 19, 37], the future human activities are predicted with different approaches in different domains. Despite previous work to understand human activities in sports and social scenes, we are more interested in *forecasting* activity by modeling the response of the players to predict changes in the scene.

American football analysis American football is one major domain of sports that has been of interest and analyzed with various approaches for different purposes for the past years [2, 5, 10, 22, 24, 32, 33, 35]. Recognition of many aspects of plays during the game of American football [2, 5, 10, 32, 33] is a long standing challenge. An algorithm for recognizing the team-based group activity of multiple players were representative in [10]. With a particular focus on the offensive plays in American football plays, an automatic framework for recognizing team formations showed a notable performance [2]. A system for recognizing different play types for a large variety of football videos was also developed recently [5]. The pattern of offensive plays (*e.g.*, trajectories of players) and the recognition of it [32, 35] is a key to understand the dynamics of American football and closely relevant to the subject dealt in this work. [20, 21] also modeled opponent players of American football successfully, which shares a conceptual idea with our work in common except that it's in an artificial game setting. Unlike previous work that focused on recognizing or detecting a specific pattern of players, we address a task of predicting for the future (*e.g.*, how a player activity will evolve over time).

Inverse reinforcement learning Recent research has also shown that predictive models can be inferred using inverse reinforcement learning[1, 3, 9, 17, 25, 28, 29]. Several works on modeling agent behavior has been presented including maximum margin based prediction [31], and combining prior knowledge and evidence to derive a probability distribution over the reward functions [29]. Also, based on the principle of maximum entropy [11] probabilistic approaches to inverse optimal control have also been developed [38, 39, 40]. Our work is most similar to [17, 38] in that we use a MDP based on the maximum entropy inverse reinforcement learning. Our main focus is, however, to enable forecasting in dynamic environment using predicted features over future unobserved states.

1.5 Contributions

The contributions of this work are summarized as follows: (1) development of short-term dynamics prediction models that accurately predict changes in the defense, (2) use of a dynamic state reward function that seamlessly integrates multiple short-term forecast distributions during inference, and (3) the application of visual activity forecasting for predicting wide receiver trajectories.

1.6 Road Map

Based on the intuition and problems explained in Section 1.1, we explain how the modeling for forecasting player activities is performed in Chapter 2. Specifically, we explain the preliminaries to represent the activity of a player using Markov decision process and reinforcement learning in Section 2.1. How static and dynamics of the environments is designed and predicted is described in Section 2.2 and 2.3 respectively. Then, using the proposed methods to predict the dynamic features (*i.e.*, defensive motion) we perform sequential inference for predicting wide receiver trajectories in Chapter 3. We present our comprehensive experimental results in Chapter 4 along with an application to sports analytics in Section 4.4.

Chapter 2

Modeling of Forecasting Environments

In this chapter, we explain background theory and how we modeled for the task of forecasting player activity. Firstly, the player’s activity is modeled using Markov decision process and reinforcement learning in Section 2.1. Motivated by the necessity of environments in reinforcement learning, we design the static environments with a prior knowledge about the game in Section 2.2. More importantly, we describe how we predict the dynamic features and environments using our regression techniques in Section 2.3.

2.1 Modeling of Player Activity

A Markov decision process (MDP) is a mathematical model for decision making process [4, 8] in a form of Markov chains defined with state s , action a , transition probability $p_{s,a}^{s'}$ and initial distribution $p(s_0)$ (see Figure 2.1). As a stochastic process, an agent takes an action to move to another state with the transition probability and accordingly takes reward $r(s)$ for that action. In our problem domain, the location of an agent (*i.e.*, wide receiver) from a top-down view in world coordinate represents the state $s = [x, y]$ and the movement to an adjacent state near the current state is defined as action a for the agent (see Figure 2.2). The goal of MDP is to find a policy π for the agent, which maximizes the function of cumulative rewards for a sequence of actions. The reward of a path $R(\zeta)$ is the sum of all state rewards $r(s)$ which is the sum of weighted feature responses $\mathcal{F}(s) \in \mathbb{R}^k$ along the path,

$$R(\zeta; \theta) = \sum_{s \in \zeta} r(s; \theta) = \sum_{s \in \zeta} \theta \cdot \mathcal{F}(s) \quad (2.1)$$

where the path ζ is a sequence of states, θ is a vector of reward weights, and the reward of a state $r(s; \theta)$ represents the immediate reward received at a state s . The feature responses $\mathcal{F}(s) = [\mathcal{F}_1(s), \mathcal{F}_2(s), \dots, \mathcal{F}_k(s)]$ are a series of different types of features where each feature response is represented as a 2D feature response map over the football field. In this work, we not only model static features, but also dynamic features to represent the high cost (*e.g.*, low reward) region which is attributed to the opponent reaction that changes over time.

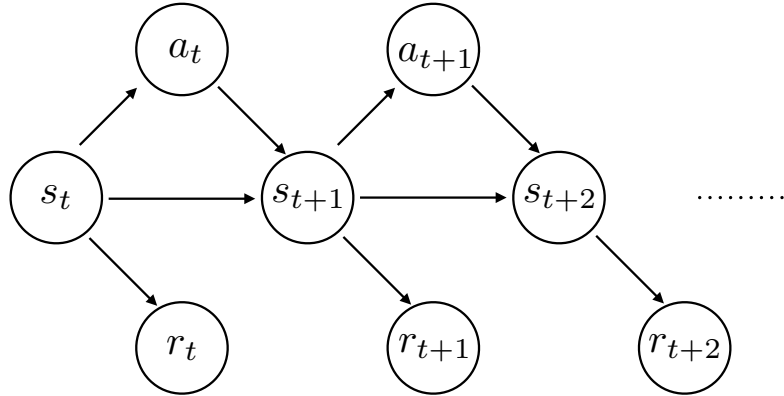


Figure 2.1: A representation of Markov decision process.

2.2 Modeling of Static Environments

In reinforcement learning, an agent constantly interacts with an environment and receives a reward as a result of its interaction (see Figure 2.3). In this process, an environment that an agent interacts with is static in general. In other words, an environment represents a state space where a state reward is defined at a specific state and does not change. In order to model the wide receiver activity, environments need to be designed, which requires a prior knowledge about the game. In this work, we utilized five features based on empirical knowledge about the game, described as the following.

- **Linear distance to the end zone:** The offense attempts to proceed forward and run towards the goal (*i.e.*, the end zone). This is represented as a linear distance along the horizontal axis.

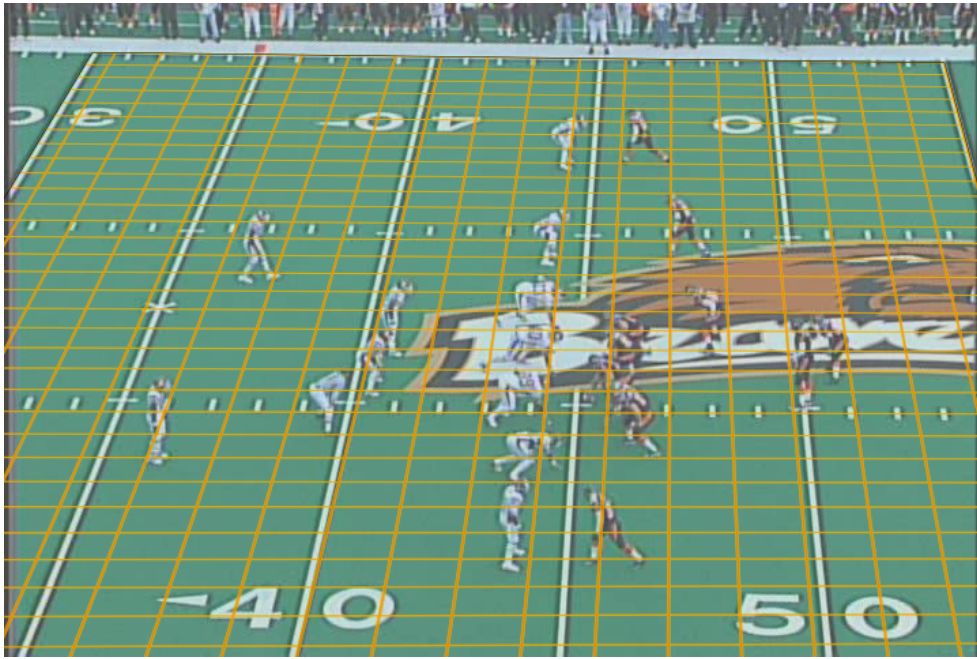


Figure 2.2: A state representation by a grid map over the football field.

- **Formation of defense:** The initial formation of the defense affects the strategic plans of the offense. For example, the wide receiver might attempt to run into empty spaces depending on the location of cornerback or safeties.
- **Route tree:** The pattern of the wide receiver activity in passing plays is described with route trees. We designed a basic route tree containing Flat, Slant, Comeback, Curl, Out, Dig, Corner, Post, Fade (see Figure 2.4).
- **A posteriori trajectory pattern:** We made use of a player trajectory dataset [35] and developed wide receiver pattern feature to the environment (see Figure 2.5).
- **Constant:** In order to prevent utility functions from being negative, we added a constant reward to all states in the static environment.

2.3 Prediction of Dynamic Environments

As each episodic play starts to evolve, the wide receiver activity is constantly affected by the defense. Therefore, the movement of defenders must be considered for a system to yield plausible forecasts for the wide receiver activity. In this work, the movement of defenders is expressed

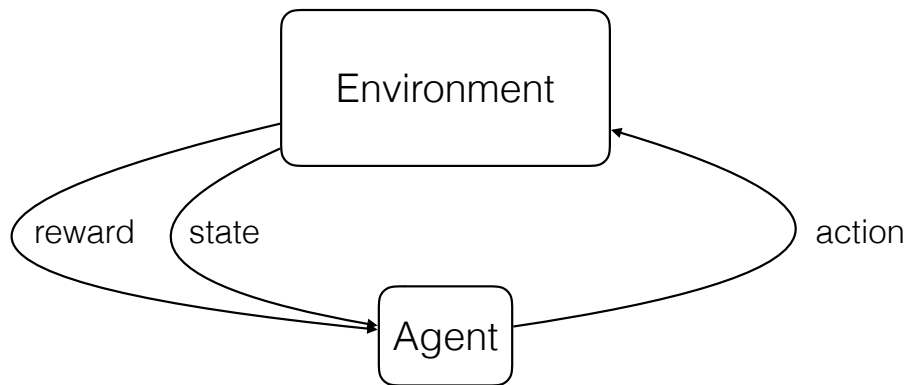


Figure 2.3: Reinforcement learning: learning takes place as a result of interaction between an agent and the environment.

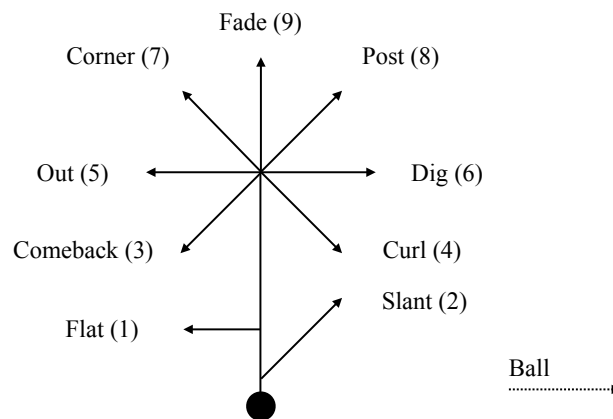


Figure 2.4: A basic route tree.

as the changes in environments, which forms dynamic environments for the wide receiver to be predicted. Since the movement of defenders (*i.e.*, where the defender will move in the future) is unknown in advance, however, it must be predicted. In order to produce reliable estimates of the future locations of the defender, two different predictors are presented: non-linear feature regressor using Gaussian process regression and linear feature regressor using constant motion.

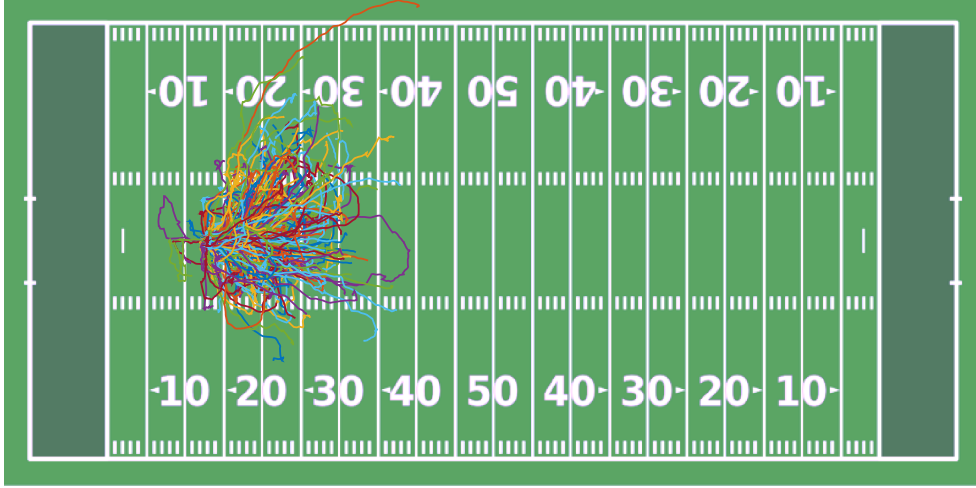


Figure 2.5: A posteriori trajectory pattern of wide receiver.

2.3.1 Non-linear feature regressor

One approach of predicting opponent reaction is to use a regressor trained in a supervised manner. Instead of assuming a simple parametric model which lacks expressive power, we propose to use a finer approach, Gaussian process regression (GPR), which gives more flexibility in representing data [30] as well as variance estimates that we use for generating dynamic features. Fully specified by its mean function $m(x)$ and covariance function $k(x, x')$, the Gaussian process generates distribution over functions f as follows,

$$f \sim \mathcal{GP}(m(x), k(x, x')) \quad (2.2)$$

where we used linear model for the mean function $m(x) = wx + b$, and an isotropic Gaussian with noise for the covariance function $k(x, x') = \sigma_y^2 \exp(-\frac{(x-x')^2}{2l^2}) + \sigma_n^2 \delta_{ii'}$. The hyperparameters $\{a, b, \sigma_y, \sigma_n, l\}$ are obtained by a gradient based numerical optimization routine. Given some noisy observations, Gaussian process regression not only fits the training examples but also computes the posterior that yields predictions for unseen test cases,

$$\mathbf{y}_* | \mathbf{y} \sim \mathcal{N}(\mu_* + \Sigma_*^T \Sigma^{-1}(\mathbf{y} - \boldsymbol{\mu}), \Sigma_{**} - \Sigma_*^T \Sigma^{-1} \Sigma_*) \quad (2.3)$$

where $\boldsymbol{\mu}$ and μ_* is the training mean and the test mean respectively, and Σ is training set covariance, Σ_* is training-test set covariance and Σ_{**} is test set covariance. For the known function values of the training cases \mathbf{y} , the prediction \mathbf{y}_* corresponding to the test inputs \mathbf{x}_* is the predicted position of the cornerback in our case.

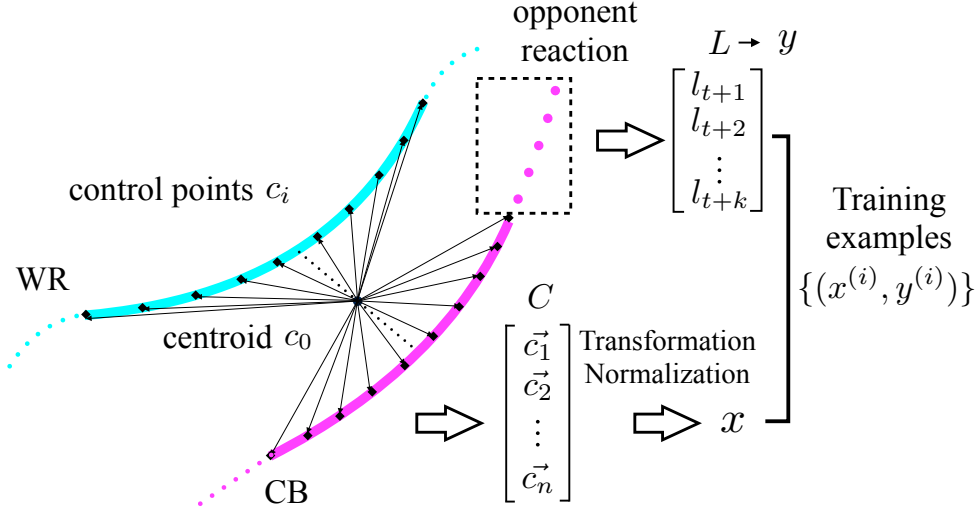


Figure 2.6: Creation of a training example used in non-linear feature regressor. Both wide receiver (cyan) and cornerback (magenta) are contained in x implicitly to predict the opponent reaction.

Given the GPR model, we collect the training samples $x^{(i)}$ and labels $y^{(i)}$ as follows. We constructed x of training examples firstly by concatenating all the relative distance vectors pointing from the centroid c_0 of the trajectory pair to the control points c_i that are uniformly picked up on the two trajectories of wide receiver and cornerback: $\vec{c}_i = \overrightarrow{c_0 c_i}$, where $i = 1, \dots, n$. Then it is rotated to a baseline to make the training example rotation-translation invariant (see Figure 2.6). Since the absolute coordinates and orientations of the trajectories vary for each scenes, this sort of data calibration is necessary. The training label y is a vectorized future locations of the cornerback for k next time steps. Having a set of N training examples of the form $\{(x^{(1)}, y^{(1)}), \dots, (x^{(N)}, y^{(N)})\}$, we created the training samples $[\mathbf{x}, \mathbf{y}]$ to perform GPR. Note that \mathbf{x} of the training examples contains the information of both wide receiver and cornerback implicitly (*i.e.*, all control points from both WR and CB) in an attempt to estimate the future opponent locations and create the dynamic features, based on the interplay between wide receiver and cornerback. Using the covariance of the prediction from the regression, we estimate the *area* (*i.e.*, 95% confidence interval) of the opponent reaction as the dynamic feature.

2.3.2 Linear (constant motion) feature regressor

Since the motion of a player is unlikely to change drastically in a short period of time, one natural approach of estimating the opponent reaction in the near future is to directly model the human activity using instantaneous speed. Specifically, the velocity of a player at current state is determined by averaging over the velocities in the past frames. Then we can estimate the location of a player in the future using the velocity at current state computed as follows,

$$l_{t+k} = l_t + \frac{k}{n_f} \sum_{i=t-n_f}^{t-1} V_i \quad (2.4)$$

where l_t is the location of a player (represented in 2D) at time t , l_{t+k} is the location of a player in k time step later from the current state, n_f is the number of past frames, and V_i is the velocity at frame i . Thus, the k next time step future location of the opponent player is estimated by k multiple of the average velocity per frame plus the current location. Depending on the number past frames used, the average velocity will be different as well as the estimated locations of a player. The estimated locations of a player is then Gaussian filtered to form the area of the opponent reaction as the dynamic feature.

Chapter 3

Forecasting in Dynamic Environments

In this chapter, we describe how forecasting is performed in dynamic environments. First, we explain how we recover the optimal reward weights using maximum entropy inverse optimal control in Section 3.1. Then, how a varying reward in dynamic environments is incorporated into a sequential inference procedure is explained in Section 3.2. The specific algorithm is elaborated in Section 3.3.

3.1 Maximum entropy inverse optimal control

As the reward weights θ is typically unknown, in inverse reinforcement learning (IRL) or inverse optimal control (IOC) it is attempted to recover the reward weights from demonstrated examples such that the agent model generates action sequences similar to a set of demonstrated examples [1]. In maximum entropy inverse reinforcement learning [38], the distribution over a path ζ is defined as,

$$p(\zeta; \theta) = \frac{e^{R(\zeta; \theta)}}{Z(\theta)} = \frac{e^{\sum_{s \in \zeta} \theta \cdot \mathcal{F}(s)}}{Z(\theta)} \quad (3.1)$$

where $Z(\theta)$ is the normalizing partition function, showing that a path with higher rewards is exponentially more preferred. With the objective of recovering the optimal reward function parameters θ^* , we perform an exponentiated gradient descent to maximize the likelihood of the maximum entropy of the observed data,

$$\theta^* = \operatorname{argmax}_{\theta} \mathcal{L}(\theta) = \operatorname{argmax}_{\theta} \sum_{\zeta} \log p(\zeta; \theta) \quad (3.2)$$

where $\mathcal{L}(\boldsymbol{\theta})$ is the log-likelihood of the observation which is the trajectory of the wide receiver in our case. The gradient of the log-likelihood is expressed by the difference between the empirical mean feature counts $\bar{\mathbf{f}}$, which is the sum of the feature counts over all demonstrated trajectories $\bar{\zeta}$ and the expected mean feature counts $\hat{\mathbf{f}}_{\boldsymbol{\theta}}$ over the sampled trajectories $\hat{\zeta}$ from forecast distribution, which is represented by state visitation frequency $D(s)$ as follows,

$$\begin{aligned}\bar{\mathbf{f}} &= \sum_{\bar{\zeta}} \sum_{s \in \bar{\zeta}} \mathcal{F}(s), \\ \hat{\mathbf{f}}_{\boldsymbol{\theta}} &= \sum_{\hat{\zeta}} p(\zeta_s; \boldsymbol{\theta}) \sum_{s \in \hat{\zeta}} \mathcal{F}(s) = \sum_s D(s) \mathcal{F}(s).\end{aligned}\tag{3.3}$$

As the expected feature count matches to the empirical feature count, the learner aims to mimic the demonstrated behavior [1].

3.2 Sequential inference in dynamic environment

In dynamic environment where dynamic features come from the opponent reaction, the state reward should change over time as players move during the play. We thus define the state reward as a linear combination of the static state reward and the dynamic state reward as the following,

$$r_t(s; \boldsymbol{\theta}) = r(s; \boldsymbol{\theta}_s) + r_t(s; \boldsymbol{\theta}_d)\tag{3.4}$$

where $r_t(s; \boldsymbol{\theta})$ represents the time-varying state reward at time t while $\boldsymbol{\theta}_s$ and $\boldsymbol{\theta}_d$ are the reward weights of the static features and the dynamic features respectively.

Unlike the standard forecasting approaches in static environments where they performed a single long-term forecasting [17], we perform multiple short-term predictions sequentially while updating the dynamic feature. Note in this process that we do not use any real-time observations to estimate the dynamic features. It is our interest to perform a long sequenced short-term forecasts based on the predicted dynamic features.

To be more precise, the forecast distribution is expressed as a state visitation frequency $D(s)$ which is computed by propagating the initial distribution $D(s_0)$ using optimal policy. In our case we have a short-term forecast distribution $\tilde{D}^{(t)}(s)$ at each time step t , which changes over time and gradually produces the final forecast distribution. In this process, we pass the previous short-term forecast distribution as input to the next inference cycle so that the forecasting proceeds without discontinuity. It is also necessary to have the predicted short-term trajectories of

the wide receiver and cornerback, $\tilde{\zeta}_{WR}$ and $\tilde{\zeta}_{CB}$ respectively, for the next inference. From the forecast distribution at each time step, the state with the maximum probability is chosen to be the predicted location for the wide receiver. The predicted locations altogether form the predicted short-term trajectory of the wide receiver $\tilde{\zeta}_{WR}$. Then combined together with the estimated opponent location, they become the input of the next forecasting cycle. This sequential inference yields the final forecast distribution which is no shorter in prediction length than the current state-of-the-art [17], but enables forecasting in dynamic environment.

3.3 Algorithm

Solving for the reward function reduces to the problem of recovering the optimal reward weight θ^* that best describes the demonstrated examples. This is achieved during the training phase, through an alternating two-step algorithm called the *backward pass* and the *forward pass*.

In the *backward pass*, we firstly generate the dynamic feature $\mathcal{F}_t(s)$ using ground-truth observations, and then iteratively compute the state-action log partition function $Q(s, a)$ and the state log partition function $V(s)$ through a value iteration [4]. The soft value $V(s)$ is the expected cost to the destination from a state s , and $Q(s, a)$ is the value at one step later after taking action a from the current state. Then the policy $\pi_\theta(a|s)$ is computed by taking the exponentiated difference between $Q(s, a)$ and $V(s)$.

In the *forward pass*, we propagate the initial distribution $D(s_0)$ using the policy obtained during the backward pass. The state visitation frequency $D(s)$, or the forecast distribution, is then computed by taking the sum of all the propagated distribution at each step. Next, the expected feature count $\hat{\mathbf{f}}_\theta$ is computed by multiplying $D(s)$ by the feature responses $\mathbf{f}(s)$. The two-step algorithm repeats until the expected feature count $\hat{\mathbf{f}}_\theta$ matches the empirical feature count $\bar{\mathbf{f}}$ using the exponentiated gradient descent, $\theta \leftarrow \theta e^{\lambda \nabla \mathcal{L}(\theta)}$, where the gradient $\nabla \mathcal{L}(\theta)$ is the difference between $\bar{\mathbf{f}}_\theta$ and $\hat{\mathbf{f}}_\theta$, and λ is the step size (Eq. 3.2 and 3.3).

During the test phase, we use the optimal reward weight θ^* learned in the training phase to compute the reward function, and the backward-forward pair repeats until the wide receiver reaches to the destination resulting in the final forecast distribution (see Algorithm 1). Unlike the standard forecasting approach which takes a single long-term inference, our algorithm is designed to perform multiple short-term forecasts incrementally while updating dynamic features

Algorithm 1 Sequential Inference in Dynamic Environment

Input: θ^* (optimal reward weight)

Output: $D(s)$: forecast distribution

- 1: $\tilde{\zeta}_{WR} \leftarrow \zeta_{WR}, \tilde{\zeta}_{CB} \leftarrow \zeta_{CB},$ and $D(s_0) \leftarrow 1$
 - 2: **repeat**
 - 3: **Backward pass:**
 - 4: Estimate $\mathcal{F}_t(s)$ with $\tilde{\zeta}_{WR}$ and $\tilde{\zeta}_{CB}$
 - 5: $r_t(s; \theta^*) = r(s; \theta_s^*) + \theta_d^* \cdot \mathcal{F}_t(s)$
 - 6: **for** $n=N, \dots, 2, 1$ **do**
 - 7: $V^{(n)}(s_{goal}) \leftarrow 0$
 - 8: $Q^{(n)}(s, a) = r_t(s; \theta^*) + E_{p_{s,a}^{s'}} [V^{(n)}(s')]$
 - 9: $V^{(n-1)}(s) = \text{soft max}_a Q^{(n)}(s, a)$
 - 10: **end for**
 - 11: $\pi_\theta(a|s) \leftarrow e^{Q(s,a)-V(s)}$
 - 12: **Forward pass:**
 - 13: **for** $n = 1, 2, \dots, N_s$ **do**
 - 14: $D^{(n)}(s_{goal}) \leftarrow 0$
 - 15: $D^{(n+1)}(s) = \sum_{s',a} p_{s',a}^s \pi_\theta(a|s') D^{(n)}(s')$
 - 16: $\tilde{\zeta}_{WR} \leftarrow \max(\tilde{D}^{(n+1)}(s))$
 - 17: **end for**
 - 18: $D(s) = \sum_n D^{(n)}(s)$
 - 19: $D(s_0) \leftarrow D(s)$
 - 20: $\hat{\mathbf{f}}_\theta = \sum_s \mathcal{F}(s) D(s)$
 - 21: **until** WR reaches to the goal
-

until it reaches to the goal to enable forecasting in dynamic environment.

Chapter 4

Experimental Results

In this chapter, we describe the experiments conducted throughout this work and show the evaluations that show the validity of our approach. Specifically, the pre-processing, dataset, feature maps and metric used are described in Section 4.1. Then, we evaluate the short-term dynamics predictors (*i.e.*, regressors used to predict the defensive motion) in Section 4.2. The forecasting performance is evaluated and presented in Section 4.3 with comparisons to the previous work. Additionally, we present the result achieved as an application of our approach for sports analytics in Section 4.4.

4.1 Setup

4.1.1 Dataset

We used the OSU Digital Scout Project’s football dataset which includes the tracking results of all 22 players with the labels of player positions [7]. Using the registration matrices provided, we performed a perspective transform to register the players from the top-down view in the football field [6] as seen in Figure 4.1. The dataset contains 20 videos of *passing plays* (each video contains only one passing play). 10 videos are selected based on its lengths (several passing plays were too short to analyze, *e.g.*, 2 seconds). We performed leave-one-out cross-validation on the selected 10 videos to prevent overfitting to a specific split of the dataset.

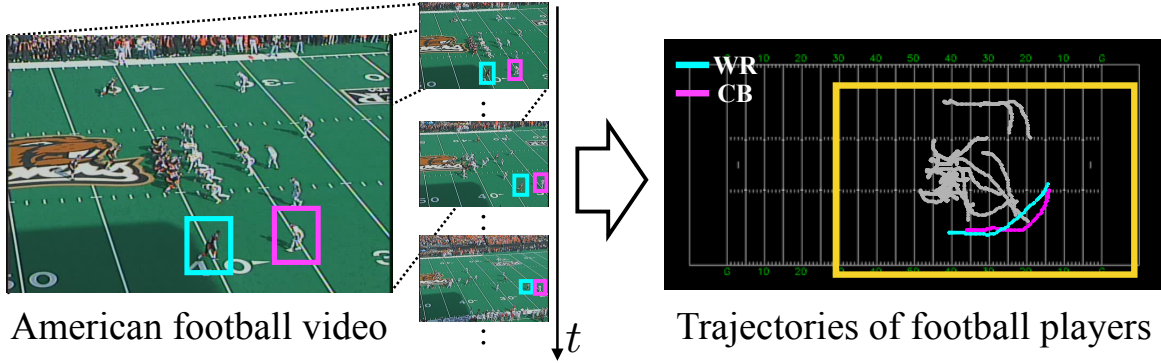


Figure 4.1: Registration of an American football video onto a top-down view ground field. Trajectory of the wide receiver (cyan) and the cornerback (magenta).

4.1.2 Feature maps

We used five types of features represented as a feature map: (1) linear distance to the end zone, (2) cornerback at the start of the play, (3) initial formation of the defense, (4) wide receiver route tree, and (5) our proposed short-term dynamics feature. These features are illustrated in Figure 4.2. The linear distance features aims to encode the logic that there is more incentive to press forward as the WR is closer to the goal line. The wide receiver route tree encodes the prior belief that the WR is more likely to follow standardized paths. The feature maps are normalized such that the reward values range from $[0, -1]$.

4.1.3 Metrics and measurements

There are 4 different metrics used to evaluate the forecasting result. First, we used the Kullback-Leibler divergence (KLD) [18] which measures the difference between two probability distributions P and Q defined by

$$D_{KL}(P||Q) = \sum_i P(i) \frac{P(i)}{Q(i)} \quad (4.1)$$

where the true data P is the ground-truth trajectory normalized by its length, $\sum_{s \in \zeta} P(s) = 1$, and the approximation Q is the forecast distribution. Second, we measured the physical distance between the trajectories sampled from the learned distribution (100 samples in our experiment) and the ground-truth trajectory in two ways: one being the Euclidean distance (L2) and the other being the modified Hausdorff distance (MHD). The MHD allows to compare the best correspond-

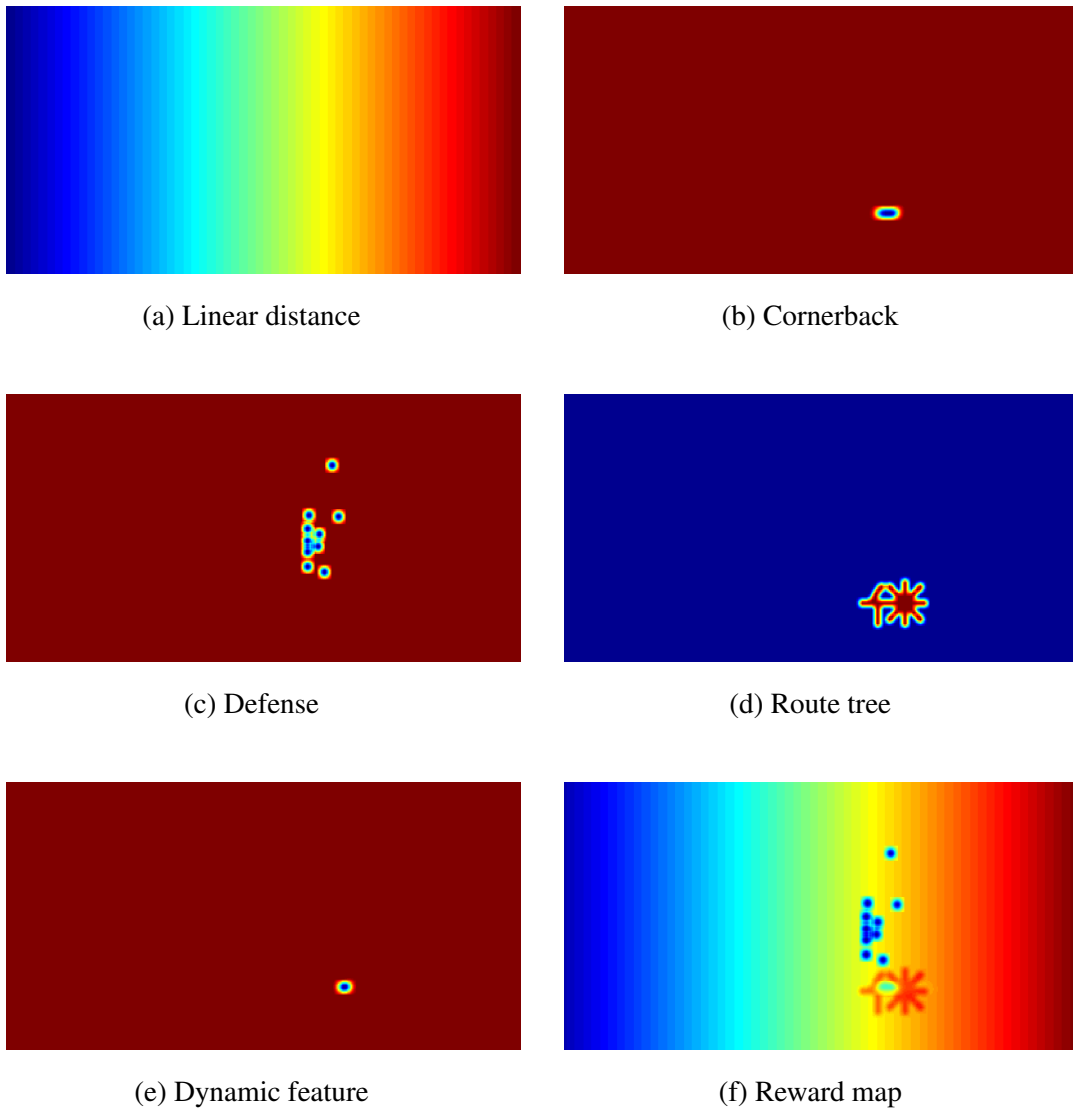


Figure 4.2: (a)-(e) Five types of feature response maps for an example scene. Each pixel of the map indicates a possible state s (represented as $[x, y]$ location in the map) that people can reach to and has a corresponding feature response value $\mathcal{F}(s)$. (f) A reward map computed using feature response maps with corresponding reward weights learned in training phase. Red region indicates the states with high reward, and the offense proceeds to the right in this scene.

ing local points within a small temporal window in two trajectories through local time warping. The computed distance is divided by the length of trajectory to accommodate trajectories of different lengths in different scenes. Furthermore, we measured the likelihood of the demonstrated trajectory under the obtained policy by computing the negative log-loss (NLL) as follows,

$$\text{NLL}(\zeta) = E_{\pi_{\theta}(a|s)} \left[-\log \prod_{s \in \bar{\zeta}} \pi_{\theta}(a|s) \right]. \quad (4.2)$$

The distribution divergence (KLD) and the physical distances (L2 and MHD) measure how much a forecast result is different from the demonstrated example. In case of the negative log-loss (NLL), we measure how likely the demonstrated example can be, under the optimal policy learned through the proposed algorithm.

4.2 Evaluation of short-term dynamics predictors

The dynamic features are the predicted (hallucinated) locations of the moving defense. The feature map has a low reward in locations on the field where there is a defensive player, as seen in Figure 4.2e. Since the forecasted trajectory of the wide receiver depends critically on the accurate prediction of the defense, it is important to evaluate the accuracy of the short-term dynamics predictors (dynamic features). The errors of three different models of the defense (the cornerback) are plotted in Figure 4.3. The first model is the naive static assumption, where the wide receiver plans a path based only on the initial location of the cornerback. The second predictive model CM assumes that the cornerback will maintain constant velocity (measured at the beginning of the play). The third predictive model GPR uses a Gaussian process regressor to incrementally predict the position of the cornerback. The GPR-based predictive model performs the best but has increasing error due to the accumulation of error over time.

We tested our GPR regressor on various settings for optimal parameter selections, including the length of input sample, the duration of prediction, and the number of control points. Figure 4.4a shows that there is only a nominal change in error with respect to the length of the input vector (*i.e.*, partially hallucinated trajectory), and we used an input of length 20 for the GPR. Figure 4.4b shows that the error increases as the duration of prediction increases (*i.e.*, it is harder to predict farther into the future). Therefore we selected a shorter prediction duration of 5. Figure 4.4c shows the effect of increasing the number of control points (points sampled from the

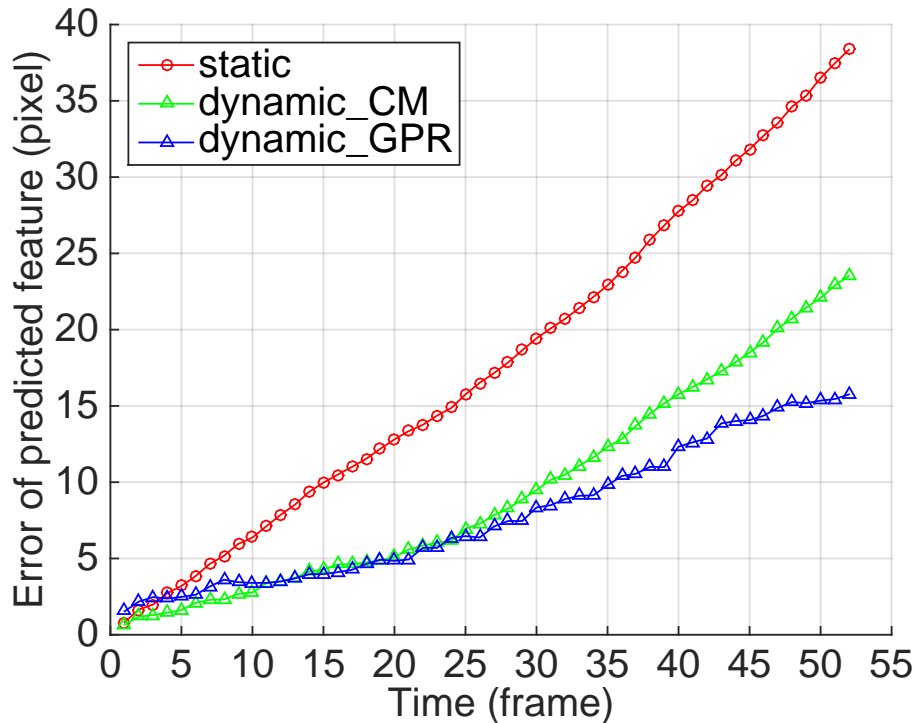
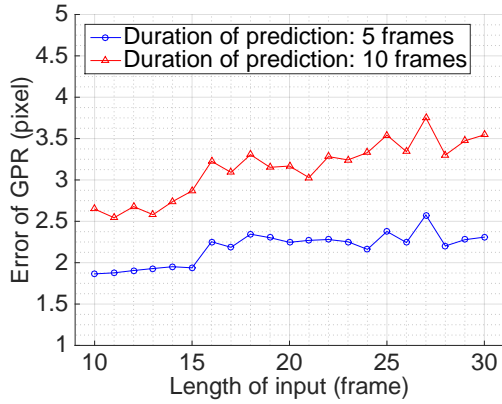


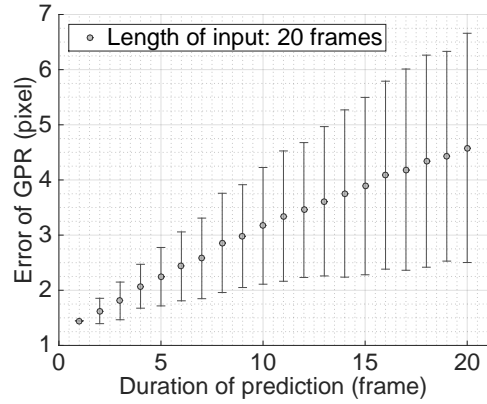
Figure 4.3: Average prediction errors of static and dynamic features. The prediction errors in dynamic environment are computed by the sequential inference procedures, while the error in static environment is computed by setting the initial location of cornerback fixed throughout the play.

hallucinated trajectories of the wide receiver and cornerback). We found that using too many control points led to overfitting, thus we chose to use only 10 control points.

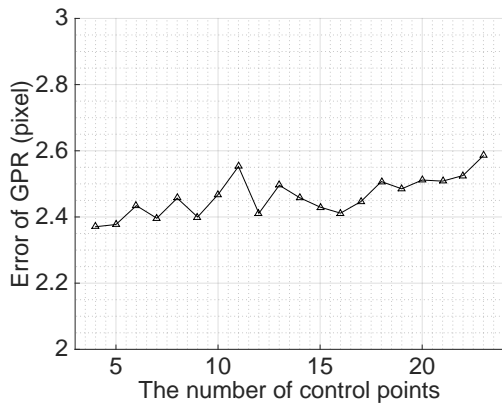
Additionally, we observed that the errors of dynamic features (*i.e.*, GPR and CM) are similar in the beginning to the middle of the game, and CM becomes more erroneous after that 4.3. The moment that GPR and CM start diverging in error is when the wide receiver and cornerback become the closest during the game as seen in 4.4d (*i.e.*, at frame 20). Note that once the wide receiver outruns the cornerback, he merely runs straight in the direction to the goal, meaning that the predictions of the opponent player is more important up until the moment he gets the closest to the wide receiver rather than in the later part of the play.



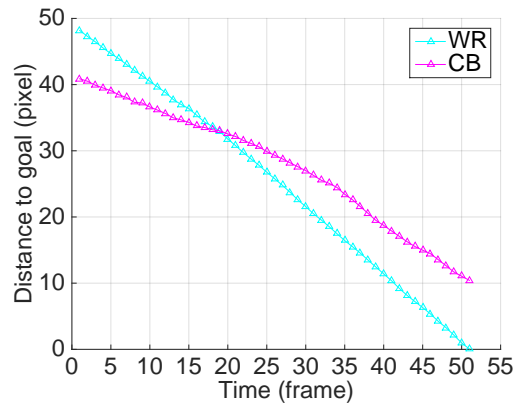
(a) Error by # of examples



(b) Error by # of predictions



(c) Error by # control points



(d) Distance to goal

Figure 4.4: (a-c) Average errors of GPR regressor in varying parameters. (d) The distance to goal for the WR and CB.

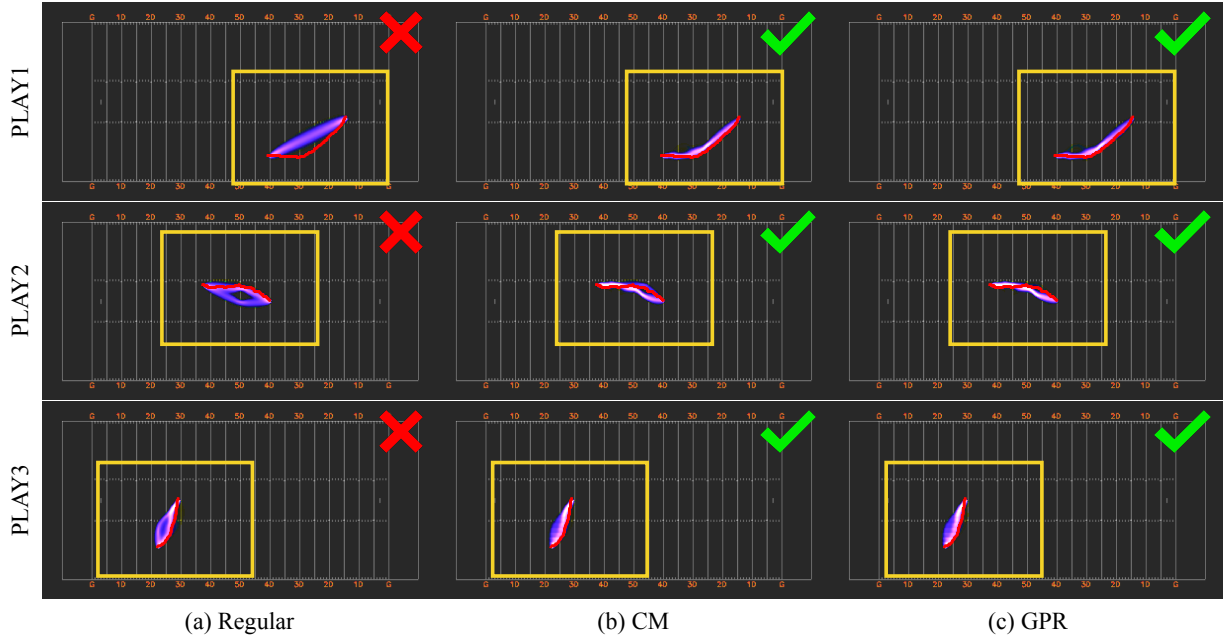


Figure 4.5: Selected forecasting results in comparison. The forecast distributions using the proposed CM (b) and the proposed GPR (c) match much closer to the ground-truth wide receiver trajectories (red) than the regular forecasting approach (a). Best viewed in electronic form.

4.3 Evaluation of activity forecasting

We evaluated the performance of our feature prediction approach over a standard forecasting framework [17] where the environment is assumed static. As expected, since we consider the changes of the environment over time, the performance gain against the standard (static feature assumption) forecasting is significant (see Figure 4.5 for a qualitative comparison). The forecast distributions (shaded area) using the proposed methods, CM and GPR, match the ground-truth trajectories (red) closely, whereas that of the static feature forecasting does not. Considering the size of the players with respect to the football field, the distribution proximity to the ground-truth using the proposed approaches are much more robust.

The forecast results are also measured quantitatively using different metrics and the average over all test scenes is summarized in the Table 4.1. As CM and GPR show little difference in average prediction errors until the wide receiver outruns the cornerback as seen in Figure 4.3 their forecast results turn out to be analogous to each other. Over all error metrics, the proposed methods outperform the current state-of-the-art [17], reducing errors by 6%, 21%, 17%, and 24%

Method	KLD	L2	MHD	NLL
Regular	9.05	9.96	8.44	2.05
Ours (CM)	8.48	7.88	7.07	1.55
Ours (GPR)	8.51	7.87	7.03	1.58

Table 4.1: Average errors of forecasting approaches measured in different metrics. The proposed methods, CM and GPR, outperform the standard forecasting model [17] in all criteria.

(in order of KLD, L2, MHD and NLL).

4.4 Application to Sports Analytics

One application of activity forecasting in the sports domain is its use for sports analytics. Forecasting distributions can be used to convey how easy it is for the opponent team to predict offensive motion. It can also be used to understand how players diverge from typical play motion. Using the forecasting framework introduced in this paper, it is possible to update the forecasting distribution while the play is in motion. A benefit of observing real data is that it allow us to hypothesize about the final goal (destination) of the wide receiver as the play unfolds. This task is called destination forecasting.

In order to infer the destination of the wide receiver as the play unfolds, we can compute the posterior distribution over goals using the ratio of partition functions as follows,

$$\begin{aligned}
 p(s_d | s_0, u_{1:t}) &\propto p(u_{1:t} | s_d, s_0) \cdot p(s_d) \\
 &\propto e^{V_{u_{1:t}}(s_d) - V(s_d)} \cdot p(s_d),
 \end{aligned}
 \tag{4.3}$$

where $V_{u_{1:t}}(s_d)$ is the state log partition function with observation $u_{1:t}$, $V(s_d)$ is the state log partition function without any observation, and s_d refers to the state at the destination. By switching the role of the start location to the destination, the value function of the potential destinations can be computed. As more observations of both the wide receiver and the opponent are revealed, the changes in the reward yield the progress toward a goal. An incremental visualization of destination forecasting is shown in Figure 4.7. It can be seen that the forecasting distribution becomes closer to the true trajectory (red) as more observations are revealed. The average KLD over all test scenes are plotted in Figure 4.6 as a quantitative measurement. We believe that this type of

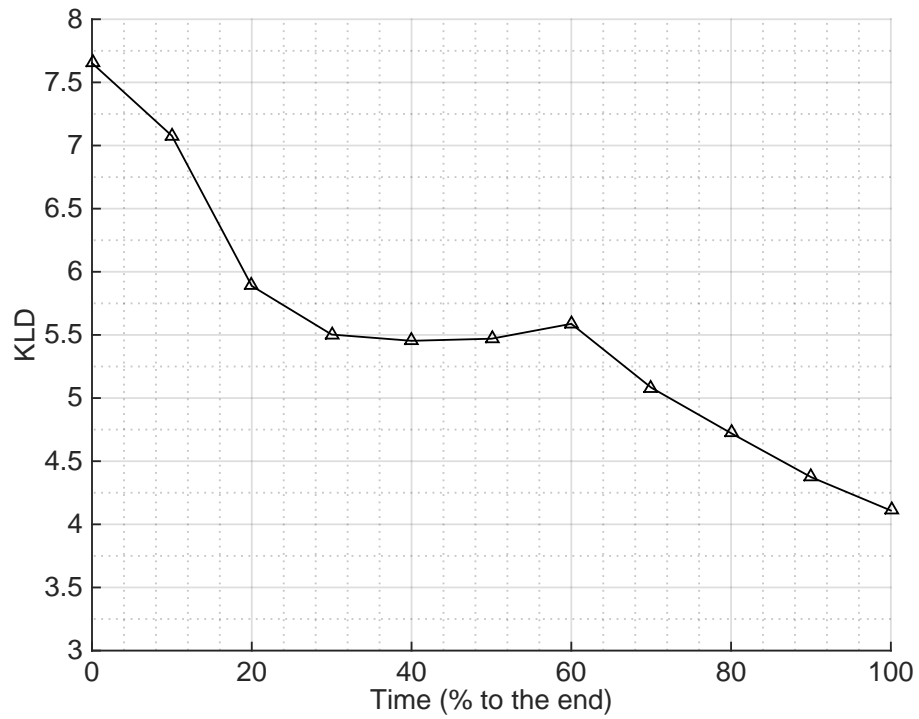


Figure 4.6: Average KLD over all test scenes from destination forecasting. The error decreases as play comes to the end.

visual prediction can be used for sport analytics.

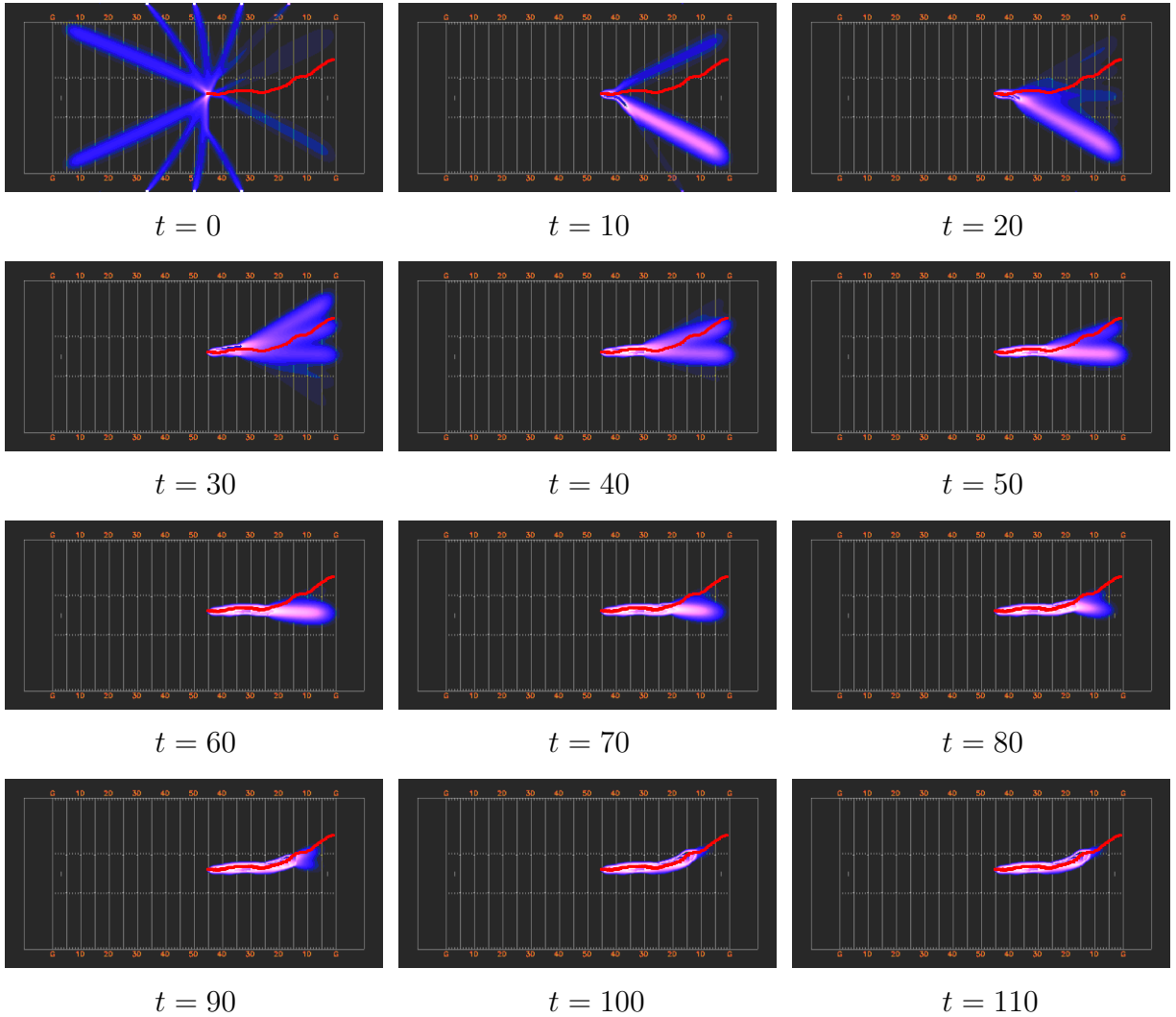


Figure 4.7: A destination forecasting result. The forecast distribution becomes closer to the demonstrated trajectory (red) as more observations are revealed by time. Best viewed in electronic form.

Chapter 5

Conclusion

We have presented a novel approach of forecasting an agent's activity in dynamic environments by focusing on the task of predicting wide receiver throwing plays in American football. The prediction of possible alterations in visual environment has been neglected and an ill-posed problem, yet we proposed to embrace it within an optimal control framework via dynamic feature regression techniques that predict opponent reactions. We demonstrated that a careful design in the strategic role of an agent, plus modeling the changes of environment in a sequential inference scheme enables successful play forecasting in dynamic environments.

As future work, we plan to develop a general feature regressors that can be applied to other player positions over the defense. Also, we are in the process of collecting data in other domain of field sports such as soccer, in order to further verify the validity of our approach for forecasting sports plays.

Bibliography

- [1] Pieter Abbeel and Andrew Y Ng. Apprenticeship learning via inverse reinforcement learning. In *Proceedings of the twenty-first international conference on Machine learning*, page 1. ACM, 2004. 1.4, 3.1, 3.1
- [2] Indriyati Atmosukarto, Bernard Ghanem, Satyajeet Ahuja, Karthik Muthuswamy, and Narendra Ahuja. Automatic recognition of offensive team formation in american football plays. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2013 IEEE Conference on*, pages 991–998. IEEE, 2013. 1.4
- [3] Chris L Baker, Rebecca Saxe, and Joshua B Tenenbaum. Action understanding as inverse planning. *Cognition*, 113(3):329–349, 2009. 1.4
- [4] Richard Bellman. Dynamic programming and lagrange multipliers. *Proceedings of the National Academy of Sciences of the United States of America*, 42(10):767, 1956. 2.1, 3.3
- [5] Sheng Chen, Zhongyuan Feng, Qingkai Lu, Behrooz Mahasseni, TS Fiez, Alan Fern, and Sinisa Todorovic. Play type recognition in real-world football video. In *Applications of Computer Vision (WACV), 2014 IEEE Winter Conference on*, pages 652–659. IEEE, 2014. 1.4
- [6] Robin Hess and Alan Fern. Improved video registration using non-distinctive local image features. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pages 1–8. IEEE, 2007. 4.1.1
- [7] Robin Hess and Alan Fern. Discriminatively trained particle filters for complex multi-object tracking. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 240–247. IEEE, 2009. 4.1.1
- [8] Ronald A Howard. Dynamic programming and markov processes.. 1960. 2.1
- [9] De-An Huang and Kris M Kitani. Action-reaction: Forecasting the dynamics of human interaction. In *Computer Vision–ECCV 2014*, pages 489–504. Springer, 2014. 1.4
- [10] Stephen S Intille and Aaron F Bobick. Recognizing planned, multiperson action. *Computer Vision and Image Understanding*, 81(3):414–445, 2001. 1.4
- [11] Edwin T Jaynes. Information theory and statistical mechanics. *Physical review*, 106(4): 620, 1957. 1.4
- [12] Yun Jiang and Ashutosh Saxena. Modeling high-dimensional humans for activity anticipation using gaussian process latent crfs. In *Robotics: Science and Systems, RSS*, 2014. 1.4

- [13] Kihwan Kim, Matthias Grundmann, Ariel Shamir, Iain Matthews, Jessica Hodgins, and Irfan Essa. Motion fields to predict play evolution in dynamic sport scenes. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 840–847. IEEE, 2010. 1.4
- [14] Kihwan Kim, Dongryeol Lee, and Irfan Essa. Gaussian process regression flow for analysis of motion trajectories. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 1164–1171. IEEE, 2011. 1.4
- [15] Kihwan Kim, Dongryeol Lee, and Irfan Essa. Detecting regions of interest in dynamic scenes with camera motions. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 1258–1265. IEEE, 2012. 1.4
- [16] Sujeong Kim, Stephen J Guy, Wenxi Liu, Rynson WH Lau, Ming C Lin, and Dinesh Manocha. Predicting pedestrian trajectories using velocity-space reasoning. In *Algorithmic Foundations of Robotics X*, pages 609–623. Springer, 2013. 1.4
- [17] Kris M Kitani, Brian D. Ziebart, J. Andrew (Drew) Bagnell, and Martial Hebert. Activity forecasting. In *European Conference on Computer Vision*. Springer, October 2012. (document), 1.1, 1.1, 1.4, 3.2, 4.3, 4.1
- [18] Solomon Kullback and Richard A Leibler. On information and sufficiency. *The annals of mathematical statistics*, pages 79–86, 1951. 4.1.3
- [19] Tian Lan, Tsung-Chuan Chen, and Silvio Savarese. A hierarchical representation for future action prediction. In *Computer Vision–ECCV 2014*, pages 689–704. Springer, 2014. 1.4
- [20] Kennard Laviers and Gita Sukthankar. A real-time opponent modeling system for rush football. In *IJCAI Proceedings-International Joint Conference on Artificial Intelligence*, volume 22, page 2476. Citeseer, 2011. 1.4
- [21] Kennard R Laviersa and Gita Sukthankarb. Using opponent modeling to adapt team play in american football. 2014. 1.4
- [22] Ruonan Li, Rama Chellappa, and S Kevin Zhou. Learning multi-modal densities on discriminative temporal interaction manifold for group activity recognition. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 2450–2457. IEEE, 2009. 1.4
- [23] Patrick Lucey, Alina Bialkowski, Peter Carr, Stuart Morgan, Iain Matthews, and Yaser Sheikh. Representing and discovering adversarial team behaviors using player roles. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 2706–2713. IEEE, 2013. 1.4
- [24] Behrooz Mahasseni, Sheng Chen, Alan Fern, and Sinisa Todorovic. Detecting the moment of snap in real-world football videos. In *IAAI*, 2013. 1.4
- [25] Gergely Neu and Csaba Szepesvári. Apprenticeship learning using inverse reinforcement learning and gradient methods. *arXiv preprint arXiv:1206.5264*, 2012. 1.4
- [26] NFL. Official playing rules and casebook of the national football league, 2015. URL <http://operations.nfl.com/the-rules/>. 1.2
- [27] Stefano Pellegrini, Andreas Ess, and Luc Van Gool. Predicting pedestrian trajectories. In

Visual Analysis of Humans, pages 473–491. Springer, 2011. 1.4

- [28] Stéphane Petti and Thierry Fraichard. Safe motion planning in dynamic environments. In *Intelligent Robots and Systems, 2005.(IROS 2005). 2005 IEEE/RSJ International Conference on*, pages 2210–2215. IEEE, 2005. 1.4
- [29] Deepak Ramachandran and Eyal Amir. Bayesian inverse reinforcement learning. *Urbana*, 51:61801. 1.4
- [30] Carl Edward Rasmussen. Gaussian processes for machine learning. 2006. 2.3.1
- [31] Nathan D Ratliff, J Andrew Bagnell, and Martin A Zinkevich. Maximum margin planning. In *Proceedings of the 23rd international conference on Machine learning*, pages 729–736. ACM, 2006. 1.4
- [32] Behjat Siddiquie, Yaser Yacoob, and Larry Davis. Recognizing plays in american football videos. Technical report, Technical report, University of Maryland, 2009. 1.4
- [33] Eran Swears and Anthony Hoogs. Learning and recognizing complex multi-agent activities with applications to american football plays. In *Applications of Computer Vision (WACV), 2012 IEEE Workshop on*, pages 409–416. IEEE, 2012. 1.4
- [34] Raquel Urtasun, David J Fleet, and Pascal Fua. 3d people tracking with gaussian process dynamical models. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 1, pages 238–245. IEEE, 2006. 1.4
- [35] Jagannadan Varadarajan, Indriyati Atmosukarto, Shaunak Ahuja, B Ghanem, and N Ahuja. A topic model approach to represent and classify american football plays. In *British machine vision conference (BMVC)*, 2013. 1.4, 2.2
- [36] Julian Walker, Arpan Gupta, and Martial Hebert. Patch to the future: Unsupervised visual prediction. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 3302–3309. IEEE, 2014. 1.4
- [37] Dan Xie, Sinisa Todorovic, and Song-Chun Zhu. Inferring” dark matter” and” dark energy” from videos. In *Computer Vision (ICCV), 2013 IEEE International Conference on*, pages 2224–2231. IEEE, 2013. 1.4
- [38] Brian D Ziebart, Andrew L Maas, J Andrew Bagnell, and Anind K Dey. Maximum entropy inverse reinforcement learning. In *AAAI*, pages 1433–1438, 2008. 1.4, 3.1
- [39] Brian D Ziebart, Nathan Ratliff, Garratt Gallagher, Christoph Mertz, Kevin Peterson, James A Bagnell, Martial Hebert, Anind K Dey, and Siddhartha Srinivasa. Planning-based prediction for pedestrians. In *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*, pages 3931–3936. IEEE, 2009. 1.4
- [40] Brian D Ziebart, J Andrew Bagnell, and Anind K Dey. The principle of maximum causal entropy for estimating interacting processes. *Information Theory, IEEE Transactions on*, 59(4):1966–1980, 2013. 1.4