Sports Field Localization via Deep Structured Models

Namdar Homayounfar Sanja Fidler Raquel Urtasun
Department of Computer Science
University of Toronto
{namdar, fidler, urtasun}@cs.toronto.edu

Abstract

In this work, we propose a novel way of efficiently localizing a sports field from a single broadcast image of the game. Related work in this area relies on manually annotating a few key frames and extending the localization to similar images, or installing fixed specialized cameras in the stadium from which the layout of the field can be obtained. In contrast, we formulate this problem as a branch and bound inference in a Markov random field where an energy function is defined in terms of semantic cues such as the field surface, lines and circles obtained from a deep semantic segmentation network. Moreover, our approach is fully automatic and depends only on a single image from the broadcast video of the game. We demonstrate the effectiveness of our method by applying it to soccer and hockey.

1. Introduction

Sports analytics is used to increase a team's competitive edge by gaining insight into the different aspects of its playing style and the performance its players. For example, sports analytics was a major component of Germany's successful World Cup 2014 campaign. Another important application is to improve scouting by identifying talented prospects in junior leagues and assessing their competitive capabilities and potential fit in a future team's roster. Sports analytics is also beneficial in fantasy leagues, giving fantasy players access to statistics that can enhance their game play. Even more impressive is the global sports betting market, which is worth up to trillion dollors according to Statista.

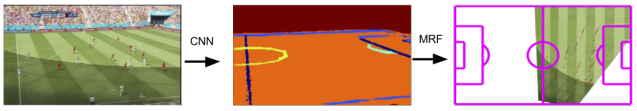
A holy grail for sports analytics is the ability to automatically extract valuable statistics from visual information alone. Being able to identify team formations and strategies as well as assessing the performance of individual players is reliant upon understanding where the actions are taking place in 3D space. This requires accurate correspondence between the playing field seen by the camera and the metric model of the field.

Most approaches to player detection [21, 27, 20, 16], game event recognition [5, 22], and team tactical analysis [18, 4, 15] perform field localization by either semi-manual methods [13, 32, 2, 31, 30, 7, 19, 1, 12] or by obtaining the game data from fixed and calibrated camera systems installed around the venue.

In this paper, we tackle the challenging task of field localization from a single broadcast image. We propose a method that requires no manual initialization and is applicable to any video of the game recorded with a single camera. Our approach bypasses the reliance on humans annotating keyframes for each new game or installing expensive cameras around the arena. The input to our system is a single image and the 3D model of the field, and the output is the mapping that takes the image to the model. In particular, we frame the field localization problem as inference in a Markov Random Field with potentials derived from a deep semantic segmentation network.

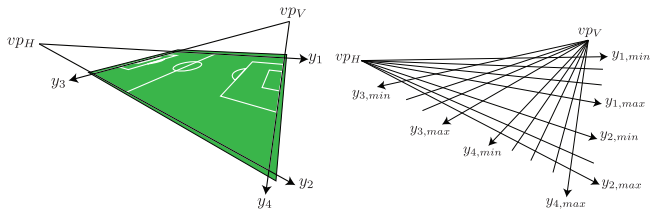
We parametrize the field in terms of four rays, cast from two orthogonal vanishing points. The rays correspond to the outer lines of the field and thus define the field's precise localization. Our MRF energy uses several potentials that exploit semantic segmentation clues such as the field surface, the line and circle markings as well as geometric agreement between the lines and circles found in the image and those defined by the known model of the field. All of our potentials can be efficiently computed. We perform inference with branch-and-bound, achieving on average less than half a second running time per frame. The weights in our MRF are learned using S-SVM [28].

For evaluation, we apply our method to the sports of soccer and hockey. A soccer game is usually held in an open stadium exposed to different weather and lighting conditions which might create difficulties in identifying the important markings of the field. Furthermore, the texture and pattern of the grass in a soccer field differs from one stadium to another. A hockey rink in comparison is mostly white and has much smaller dimensions compared to a soccer field. On the other hand, there are usually superimposed advertisements and texts on the rink which are different from one arena to another. Our deep semantic segmentation network learns to filter out all these different sources of noise and



(a)

Figure 1: We obtain semantic segmentation of the field which is fed as evidence for fast localization into an MRF with geometric priors.



(a) (b)

Figure 2: (a) Field parametrization in terms of 4 rays y_i. (b) The grid

provide strong evidence to be used in the MRF inference. Some examples are shown in Figure 7. We note however that our method is sports agnostic and is easily extendable as long as the sport venue has known dimensions and primitive markings such as lines and circles.

For soccer, we collected a dataset of images taken from 20 different games played in the World cup 2014. We also test on an annotated hockey dataset collected by Sportlogiq, a sports analytics company based in Canada. We show that our approach significantly outperforms all baselines, while our ablation study shows the importance of all our model's components. In the following, we start with a discussion of related literature, and then describe our method.

2. Related Work

A variety of approaches have been developed in industry and academia to tackle the field localization problem. In the industry, companies such as Pixelot and Prozone have proposed a hardware approach to field localization by developing advanced calibrated camera systems that are installed in a sporting venue. This requires expensive equipment, which is only possible at the highest performance level. Alternatively, companies such as Stathleates rely entirely on human workers for establishing the homography between the field and the model for every frame of the game.

In the academic setting, the common approach to field registration is to first initialize the system by either searching over a large parameter space (e.g. camera parameters) or by manually establishing a homography for various representative keyframes of the game and then propagating this homography throughout the consecutive frames. In order to avoid accumulated errors, the system needs to be reinitialized by manual intervention. Many methods have been developed which exploit geometric primitives such as lines and/or circles to estimate the camera parameters[13, 32, 2, 31, 30]. These approaches rely on hough transforms or RANSAC and require manually specified color and texture heuristics.

An approach to limit the search space of the camera parameters is to find the two principal vanishing points corresponding to the field lines [10, 9] and only look at the lines and intersection points that are in accordance with these vanishing points and which satisfy certain cross ratios. The efficacy of the method was demonstrated only on goal areas where there are lots of visible lines. However, this approach faces problems for views of the centre of the field, where there are usually fewer lines and thus one cannot estimate the vanishing point reliably.

In [6], the authors proposed an approach that matches images of the game to 3D models of the stadium for initial camera parameter estimation [6]. However, these 3D models only exist in well known stadiums, limiting the applicability of the proposed approach.

Recent approaches, applied to Hockey, Soccer and American Football [7, 19, 1, 12] require a manually specified homography for a representative set of keyframe images per recording. In contrast, in this paper we propose a method that only relies on images taken from a single camera. Also no temporal information or manual initialization is required. Our approach could be used, for example in conjunction with [7, 19] to produce automatically smooth high quality field estimates of video.

3. 3D Field Registration

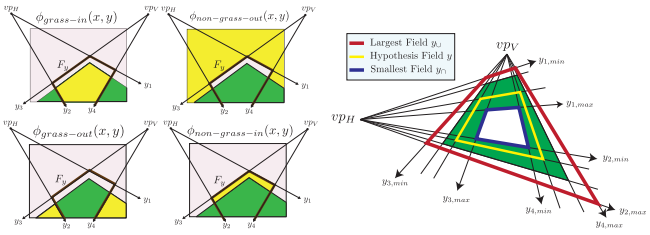
The goal of this paper is to automatically compute the transformation between a broadcast image of a sports field, and the 3D geometric model of the field.

In this section, we first show how to parameterize the problem by making use of the vanishing points, reducing the effective number of degrees of freedom to be estimated. We then formulate the problem as energy minimization in a Markov random field that encourages agreement between the model and the image in terms of field semantic segmentation cues as well as the location of the primitives (i.e., lines and circles) that mark the field. Furthermore, we show that inference can be solved exactly and very efficiently via the branch and bound algorithm.

3.1. Field Model and Parameterization

Assuming that the ground is planar, the field can be represented by a 2D rectangle embedded in 3D space. The rectangle can be defined by two long horizontal line segments and two shorter vertical line segments. Each field has also a set of vertical and horizontal lines as well as circular shapes defining different zones in the game.

The transformation between the field in the broadcast image and our 3D model can be parameterized with a homography H, which is a 3×3 invertible matrix defining a bijection that maps lines to lines between 2D projective



(a) (b)

Figure 3: (a) In each plot, the green area corresponds to grass and grey area to non-grass pixels. Field F_y is the region inside the highlighted lines. The yellow region is the percentage of counted grass/non-grass pixels. (b) The red line is the largest possible field and the blue line is the smallest field.

spaces [8]. The matrix H has 8 degrees of freedom and encapsulates the transformation of the broadcast image to the field model. One way to estimate this homography matrix is to detect points and lines in the image and associating them with points and lines in the model. Given these correspondences, the homography can be estimated in closed form using the Direct Linear Transform (DLT) algorithm [8]. While a closed form solution is very attractive, the problem lies in the fact that the association of lines/points between the image and the model is not known a priori. Thus, in order to solve for the homography, one needs to evaluate all possible assignments. As a consequence DLT-like algorithms are typically used in the scenario where a nearby solution is already known (from a keyframe or previous frame), and search is done over a small set of possible associations.

In this paper, we take a very different approach, which jointly solves for the association and the estimation of the homography. Towards this goal, we first reduce the effective number of degrees of freedom of the homography. In an image of the field, parallel lines intersect at two orthogonal vanishing points. By estimating the vanishing points, we reduce the number of degree of freedom from 8 to 4. We defer the discussion about the VP estimation to Sec. 6.

For convenience of presentation, we refer to the lines parallel to the touchlines as horizontal lines, and the lines parallel to the goallines as vertical lines. Let x be an image of the field. Denote by vp_V and vp_H the (orthogonal) vertical and horizontal vanishing points respectively.

We define a hypothesis field by four rays emanating from the vanishing points. The rays y_1 and y_2 originate from vp_H and correspond to the touchlines. Similarly, the rays y_3 and y_4 originate from vp_V and correspond to the goallines. As depicted in Fig. 2, a hypothesis field is constructed by the intersection of the four rays. Let the tuple $y = (y_1, \ldots, y_4) \in \mathcal{Y}$ be the parametrization of the field, where we have discretized the set of possible candidate rays. Each ray y_i falls in an interval $[y^{init}_{i,min}, y^{init}_{i,max}]$ and $\mathcal{Y} = \prod_{i=1}^{4} \left\{[y^{init}_{i,min}, y^{init}_{i,max}]\right\}$ is the product space of these four integer intervals. Thus $\mathcal{Y}$ corresponds to a grid.

3.2. Field Estimation as Energy Minimization

We parameterize the problem of field localization as the one of inference in a Markov random field. In particular, given an image x of the field, we obtain the best prediction $\hat{y}$ by solving the following inference task:

$$\hat{y} = \arg\max_{y \in \mathcal{Y}} w^T \phi(x, y) \tag{1}$$

with $\phi(x, y)$ a feature vector encoding various potential functions and w the set of corresponding weights which we learn using structured SVMs [28]. In particular, our energy defines different potentials encoding the priors that the field should contain mostly field surface pixels, and high scoring configurations prefer the projection of the field primitives (i.e., lines, circles) to be aligned with the detected primitives in the image (i.e. detected line segments, conic edges).

In the following we discuss the potentials in more detail.

Field Surface Potential exploits the fact that the playing field has distinguishing appearance. For example a soccer field is made of grass and a hockey rink is white ice.

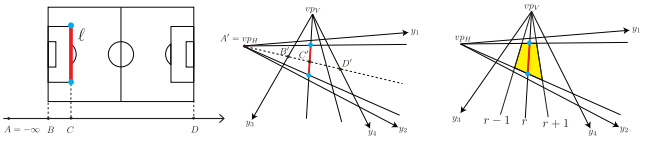
Given a hypothesis field y, let F_y denote the field restricted to the image x. We would like to maximize the number of field surface pixels in F_y. Hence, we define a potential function, denoted by $\phi_{surface-in}(x, y)$, that counts the percentage of total surface pixels that fall inside the hypothesis field F_y. However, note that for any hypothesis y' with $F_y \subset F_{y'}$, $F_{y'}$ would have at least as many surface pixels as F_y. This introduces a bias towards hypotheses that correspond to zoom-in cameras. We thus define three additional potentials that minimize the number of surface pixels outside the field F_y and the number of non-surface pixels inside F_y, while maximizing the number of non-surface pixels outside F_y. We denote these potentials as $\phi_{surface-out}(x, y)$, $\phi_{non-surface-out}(x, y)$ and $\phi_{non-surface-in}(x, y)$ respectively. We refer the reader to Fig. 3 for an illustration.

Lines Potentials: The observable lines defining the different playing zones of the field provide strong clues on the location of the sidelines. This is because their positions and lengths must always adhere to some known specifications.

We define a scoring function $\phi_\ell(x, y)$ for each line segment ℓ to yield high values when the image evidence agrees with the predicted line position obtained by reprojecting the model using the hypothesis y. The exact reprojection can be easily obtained by using the invariance property of cross ratios [8] as depicted in Fig. 4(a) in case of soccer.

Given the exact position of a line segment ℓ on the grid $\mathcal{Y}$, the score $\phi_\ell(x, y)$ counts the percentage of line segment pixels that are aligned with their corresponding vanishing point, Fig. 4(b).

Circle Potentials: A sports field usually has markings corresponding to circular shapes. When the geometric



(a) (b)

Figure 4: (a) For line ℓ (red) in the model, the cross ratio $CR = BD/BC$ must equal the cross ratio of the projection of ℓ on the grid given by $C'R' = (A'C' \cdot B'D')/(BC' \cdot A'D')$. The projection of the endpoints of ℓ are computed similarly. (b) For vertical line ℓ, the potential $\phi_\ell(x, y)$ counts the percentage of vp_V line pixels in the yellow region for which the vertical sides are one ray away from the ray on which ℓ falls upon.

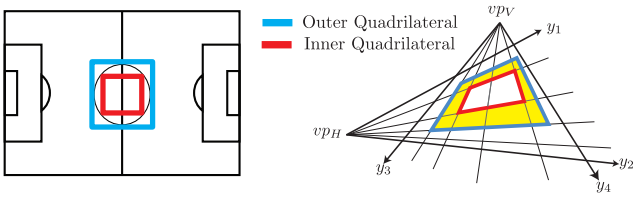
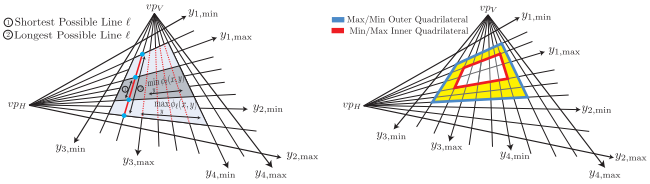


Figure 5: For each circle C in the model, the projections of the inner (red) and outer (blue) quadrilaterals can be obtained using cross ratios. Potential $\phi_C(x, y)$ is the percentage of non-vp line pixels in the yellow region.



(a) (b)

Figure 6: (a) Finding the lower and upper bounds for a line correspond to the min and max operations. (b) Upper/lower bound for $\phi_{C_i}(x, y)$ is the percentage of non-vp line pixels in the yellow region which is restricted by the max/min outer quadrilateral and the min/max inner quadrilateral.

model of the field undergoes a homography, these circular shapes transform to conics in the image.

Similar to the line potentials, we seek to construct potential functions that count the percentage of supporting pixels for each circular shape given a hypothesis field y. Unlike the projected line segments, the projected circles are not aligned with the grid $\mathcal{Y}$. However, as shown in Fig. 5 for soccer, we note that there are two unique inner and outer rectangles for each circular shape in the model which transform in the image x to quadrilaterals aligned with the vanishing points. Their position in the grid can be computed similarly as lines using cross ratios. We define a potential $\phi_C(x, y)$ for each conic as the percentage of circular pixels inside the region defined by the two quadrilaterals.

4. Exact Inference via Branch and Bound

Note that the cardinality of our configuration space $\mathcal{Y}$, i.e. the number of hypothesis fields, is of the order $O(N_H^2 N_V^2)$, which is a very large number. Here, we show how to solve the inference task in Eq. (1) efficiently and exactly. Towards this goal, we design a branch and bound [14] (BBound) optimization over the space $\mathcal{Y}$ of all parametrized fields. We take advantage of generalizations of integral images to 3D [24] to compute our bounds very efficiently.

We next explain how BBound works. Suppose that $Y \subset \mathcal{Y}$ is an arbitrary subset of parametrized fields. The priority queue of the BBound algorithm is initialized with a single set containing all the field hypotheses, i.e., $Y = \mathcal{Y}$, along with a valid upper bound $\bar{f}(Y)$. The algorithm then proceeds iteratively by taking the top element of the priority queue and splitting the set for that element into two disjoint sets. These two sets are then inserted in the priority queue. The algorithm terminates when there is a single hypothesis on top of the priority queue. For this to be true, the bounding function has to satisfy $\bar{f}(Y) = f(Y)$ when $|Y| = 1$.

Our BBound algorithm requires three key ingredients:

1. A branching mechanism that can divide any set into two disjoint subsets of parametrized fields.
2. A set function $\bar{f}$ such that $\bar{f}(Y) \geq \max_{y \in Y} w^t \phi(x, y)$.
3. A priority queue which orders sets of parametrized fields Y according to $\bar{f}$.

We next describe the first two components in detail.

4.1. Branching

Suppose that $Y = \prod_{i=1}^{4} [y_{i,min}, y_{i,max}] \subset \mathcal{Y}$ is a set of hypothesis fields. At each iteration of the branch and bound algorithm we need to divide Y into two disjoint subsets Y_1 and Y_2 of hypothesis fields. This is achieved by dividing the largest interval $[y_{i,min}, y_{i,max}]$ in half and keeping the other intervals the same.

4.2. Bounding

We need to construct a set function $\bar{f}$ that upper bounds $w^T \phi(x, y)$ for all $y \in Y$ where $Y \subset \mathcal{Y}$ is any subset of parametrized fields. Since all potential function components of $\phi(x, y)$ are positive proportions, we decompose $\phi(x, y)$ into potential with strictly positive weights and those with weights that are either zero or negative:

$$w^T \phi(x, y) = w_{neg}^T \phi_{neg}(x, y) + w_{pos}^T \phi_{pos}(x, y) \qquad (2)$$

with w_{neg}, w_{pos} the vector of negative and positive weights.

We define the upper bound on Eq. (2) to be the sum of an upper bounds on the positive features and a lower bound on the negative ones,

$$\bar{f}(Y) = w_{neg}^T \bar{\phi}^{neg}(x, Y) + w_{pos}^T \bar{\phi}^{pos}(x, Y) \qquad (3)$$

It is trivial to see that this is a valid bound. In what follows, we construct a lower bound and an upper bound for all the potential functions of our energy.

Bounds for the Field Surface Potentials: Let $y_{\cap} := (y_{1,max}, y_{2,min}, y_{3,max}, y_{4,min})$ be the smallest possible field in Y, and let $y_{\cup} := (y_{1,min}, y_{2,max}, y_{3,min}, y_{4,max})$ be the largest. We now show how to construct the bounds for $\phi_{surface-in}(x, y)$, and note that one can construct the other surface potential bounds in a similar fashion. Recall that $\phi_{surface-in}(x, y)$ counts the percentage of surface pixels inside the field. Since any possible field $y \in Y$ is contained within the smallest and largest possible fields $y_{\cap}$ and $y_{\cup}$ (Fig. 3b), we can define the the upper bound as the percentage of surface pixels inside the largest possible field and the lower bound as the percentage of surface pixels inside the smallest possible field. Thus:

$$\begin{aligned} \bar{\phi}^{pos}_{surface-in}(x, Y) &= \phi_{surface-in}(x, y_{\cap}) \\ \bar{\phi}^{neg}_{surface-in}(x, Y) &= \phi_{surface-in}(x, y_{\cup}) \end{aligned} \tag{4}$$

We refer the reader to Fig. 3(b) for an illustration.

Bounds for the Line Potentials: We compute our bounds by finding a lower bound and an upper bound for each line independently. Since the method is identical for all the lines, we will illustrate it only for the left vertical penalty line ℓ of (Fig. 4a) in case of soccer. For a hypothesis set of fields Y, we find the upper bound $\bar{\phi}^{pos}_{\ell}(x, Y)$ by computing the maximum value of $\phi_{\ell}(x, y)$ in the horizontal direction (i.e. along the rays from vp_V) but only for the maximal extended projection of ℓ in the vertical direction (i.e. along the rays from vp_H). This is demonstrated in (Fig. 6a). Finding a lower bound consists instead of finding the minimum $\phi_{\ell}(x, y)$ for minimally extended projections of ℓ.

Note that for a set of hypothesis fields Y, this task requires a linear search over all the possible rays in the horizontal (for vertical lines) at each iteration of BBound. However, as the branch and bound continues, the search space becomes smaller and finding the maximum becomes faster.

Bounds for the Circle Potentials: Referring back to the definition of the circle potentials $\phi_C(x, y)$ provided in section 3.2 and a set of hypothesis fields Y, we aim to construct lower and upper bounds for each circle potential. For an upper bound, we simply let $\phi^{pos}_C(x, Y)$ be the percentage of circle pixels contained in the region between the smallest inner and largest outer quadrilaterals as depicted in (Fig. 6b). A lower bound is obtained in a similar fashion.

4.3. Integral Accumulators for Efficient Potentials and Bounds

We construct 2D accumulators corresponding to the field surface pixels, horizontal line pixels, vertical line pixels, and circle pixels. In contrast to [29], and in the same spirit of [24], our accumulators are aligned with the two orthogonal vanishing points and count the fraction of features in the regions of x corresponding to quadrilaterals restricted by two rays from each vanishing point.

The computation of a potential function over any region in $\mathcal{Y}$ thus boils down to four accumulator lookups. Since we defined all the lower and upper bounds in terms of their corresponding potential functions, we use the same accumulators to compute the bounds in constant time.

4.4. Learning

We use structured support vector machine (SSVM) to learn the parameters w of the log linear model. Given a dataset composed of training pairs $\{x^{(n)}, y^{(n)}\}_{i=1}^{N}$, we obtain w by minimizing the augmented loss of [28] in which we have a regularization parameter $C > 0$ and a loss fuction $\Delta : \mathcal{Y} \times \mathcal{Y} \rightarrow \mathbb{R}^+ \cup \{0\}$ that measures the distance between the ground truth labeling $y^{(n)}$ and a prediction $\hat{y}$, with $\Delta(y^{(n)}, y) = 0$ if and only if $y = y^{(n)}$. In particular, we employ the parallel cutting plane implementation of [23].

The loss function is defined very similarly to $\phi_{surface-in}(x, y)$. Here, we segment the grid $\mathcal{Y}$ to field vs. non-field cells by reprojecting the ground truth field into the image.

Then given a hypothesis field y, we define the loss for a training instance $(x^{(n)}, y^{(n)})$ to be similar to the field surface potential where instead of field surface pixels we consider real field vs. non-real field in cell in the grid $\mathcal{Y}$. As a consequence, this loss can be computed using integral accumulators, and loss augmented inference can be performed efficiently and exactly using our BBound.

5. Semantic Segmentation

Our method relies on detecting important features of the field such as the field surface, lines and circles in the presence of noise. For example in soccer, we have to deal with the different textures and patterns of grass in different stadiums as well as different lighting conditions. Moreover, when detecting line and circle pixels, one has to deal with the spurious edges due to players and their shadows. Also, lots of soccer games are played in daytime in the presence of shadows which changes the color of parts of the grass and creates random edges. Most of the existing approaches use heuristics based on color and hue information to obtain these features which in turn might hinder generalization to unseen circumstances.

In this work, we opt for a simpler solution by training a semantic segmentation network that reliably detects these important features. We create GT segmentation labels for our images by using their ground truth homographies. For

soccer, we have six classes of vertical lines, horizontal lines, side circles, middle circle, grass and crowd. For hockey, we specify nine classes of vertical lines, upper horizontal sideline, lower boundary between the crowd and the rink, middle circle, face off spots and circles, the rink, the crowd, and the 4 quarter circles cornering the field. Some of the segmentation results are shown in Figure 7.

For our network, we take the trained 16-layer VGG network [25] and keep the first 7 convolution layers. We remove the pooling operations in these layers but use dilated convolutions [33] to keep the dimensions of the output layers the same as the input image. We add 5 extra convolution layers with the first three layers being 3x3 dilated convolutions. The last two layers have 3x3 and 1x1 filters without any dilation. Each added layer would have L output channels where L is the number of pixel categories. We use batch normalization after each layer and apply Relu nonlinearities throughout the network. The final output layer, which is the same dimension as the input, would give a softmax score for the category of each pixel. We minimize the cross entropy loss of each pixel in order to learn the weights. We skipped downsampling to keep the global structure of the field and used dilated convolutions in order to have bigger receptive fields.

One difficulty of learning in this task is the class imbalance of the lines and circle contours with respect to the field surface and the crowd. For instance, each ground truth line segment would have a width of 1 pixel. We tackled this in two ways: First, we artificially dilated each ground truth line segment to have a width of 10 pixels. Thus our ground truth would be a region around the line segment as opposed to the line segment alone. Second, we modified the cross entropy loss of each pixel to be $L = \sum_{\ell=1}^{L} \hat{q}_\ell \log q_\ell^{s_\ell}$. where $\hat{q}$ and q are the ground truth and score of the pixel respectively and s_ℓ is a fixed penalty term for rare categories. We finally train the network with an initial learning rate of 0.01 and the RMSProp optimizer [26] until the mean IOU on a validation set stops increasing. The network was trained on a single GPU of DGX1 and it took almost day for both soccer and hockey.

6. Vanishing Point Estimation

In a Manhattan world, such as a soccer stadium or a hockey arena, there are three principal orthogonal vanishing points. Our goal is the find the two orthogonal vanishing points vp_V and vp_H that correspond to the vertical and horizontal lines on the field. Since we know which pixels belong to the vertical and horizontal lines from our semantic segmentation network, we fit line segments to these pixels and deploy the line voting procedure of [11] to find the vanishing points. This procedure is robust when there are enough clues for each vanishing point. That is if the camera is zooming on a segment of the field where there are no line markings present, for example we only see grass, we cannot find the vanishing points without temporal context from previous frames. This is a fair assumption since this would be a difficult task even for a human.

For soccer in particular, the field is very large and sometimes the camera faces the centre of the field where there are not enough line segments to find the vertical vanishing point. In this cases, we take the line segments that belong to neither vanishing point and fit an ellipse [3] which is an approximation to the conic in the centre of the field. We then take the 4 endpoints of the ellipses' axes and also one additional point corresponding to the crossing of the ellipses' minor axis from the grass region to non-grass region to find an approximate homography which in turn gives us an approximate vp_V. For hockey, we exploit our large dataset and also the fact that hockey arenas are very similar as compared to soccer stadiums and we guide the vp estimation and grid creating in each image as follows: First, we retrieve the image's nearest neighbour in the training set based on the distance transform [17] distance of their semantic segmentations. Then, we only look for vote for vanishing points in a small region around the vanishing points of the nearest neighbour image. We also restrict our grid such that each interval in $\mathcal{Y}$ is 40 rays around the ground truth sidelines of the nearest neighbour image. This would correspond to 40^4 unique states for our search space.

	G	L	C	Mean±Sd Val IOU	Mean Test IOU
Shared	✓	✓		0.83±0.017	0.79
		✓	✓	0.88±0.016	0.84
	✓	✓	✓	0.88±0.016	0.83
Not Shared	✓	✓	✓	0.88±0.01	0.83

Table 1: **G** correspond to 4 weights for each grass potential. **L**: all the lines share the same weight. **C**: all the circles share the same weight. **Shared** means the lines **L** and the circles **C** have shared weights. **Not Shared** means that the vertical lines have different weights than the horizontal lines and also each circle has its own weight

Method	Soccer	Hockey One	Hockey All
Field NN	0.68	0.70	0.80
Semantic Seg NN	0.73	0.73	0.81
Ours	**0.83**	**0.81**	**0.82**

Table 2: Comparison with baselines. Hockey One corresponds to the experiment with only one game and Hockey All to the experiment with all the games.

Sports	Mean Time (s)	Mean Iter	# of States
Soccer	0.44	3328	$300^2 \times 600^2$
Hockey All	0.04	565	$40^2 \times 40^2$

Table 3: Inference time and number of iterations for branch and bound

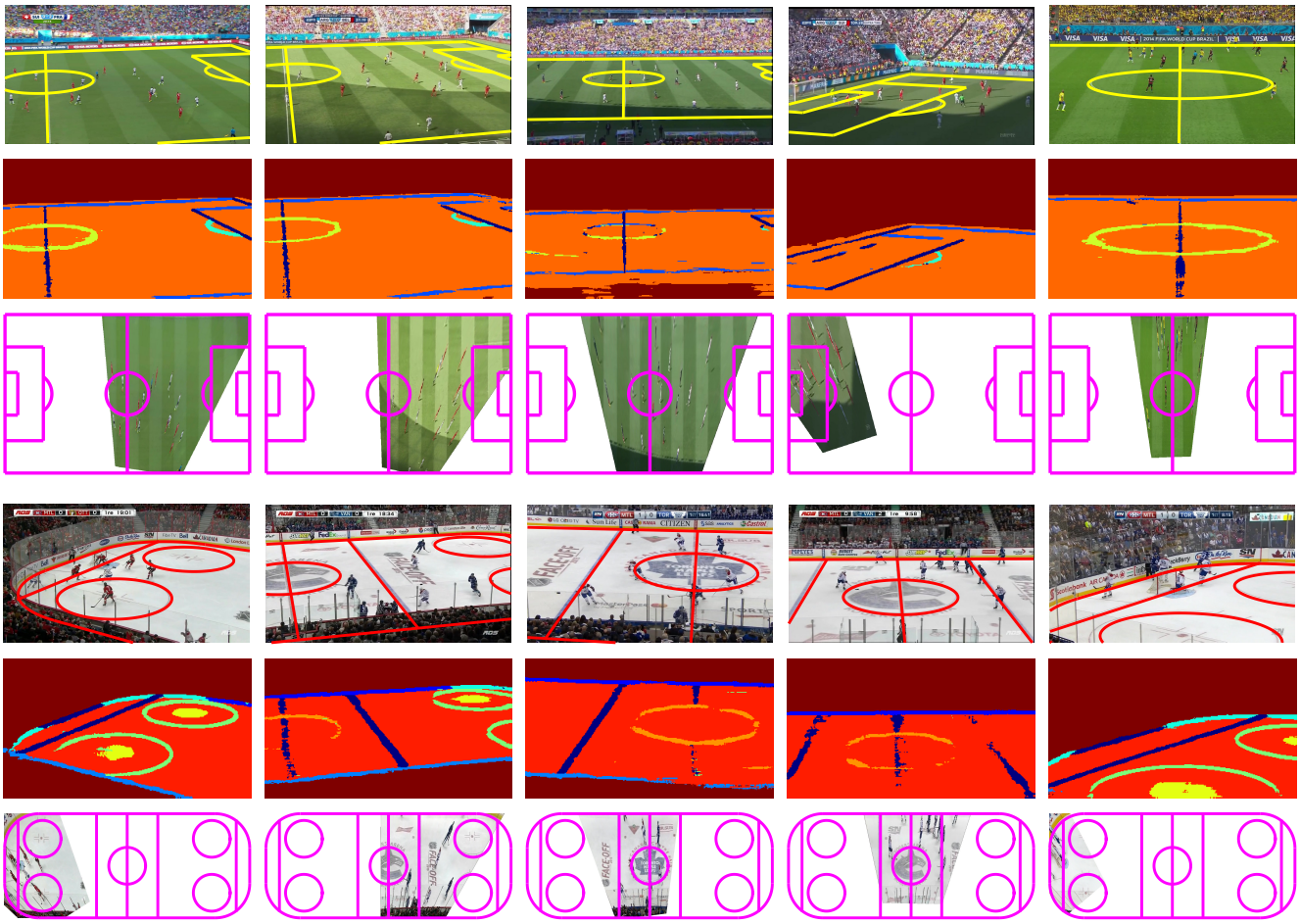


Figure 7: Examples of the obtained homographies and semantic segmentations.

7. Experiments

We apply our method to soccer and hockey. For soccer, we recorded 20 games from the World Cup 2014 held in Brazil. Out of these games we annotated 395 images with the ground truth fields and also the grass segmentations. We randomly split the games into two sets with 10 games of 209 images for training and validation, and 186 images from 10 other games for the test set. We artificially augmented the training set by flipping each image horizontally about its centre. Thus we used 418 images for training. The images consist of different views of the field with different grass textures and lighting patterns. These games were held in 9 unique stadiums during day and night. There are some games with rain and heavy shadows. We remind the reader that these images do not have a temporal ordering. In what follows we assess different components of our method. We will release this dataset to the public.

For hockey, we obtained eight fully annotated games from SportLogiq, a Canadian sports analytics company. These 8 games are played on 7 different arenas. We conduct two different experiments. First, we randomly divide the games to training and test games and randomly choose 2000 images for each set with 500 images from each game. For the second hockey experiment, we pick one of the games in the test set, and randomly split it to two sets of respective sizes of 50 and 450. We apply the learnt segmentation network of our large dataset to obtain semantic labels for these images. We then learn from scratch the weights of the MRF using these 50 examples and evaluate the mean IOU on the other 450 images. The game and its arena are different than those in the training set and as such the application of the segmentation network is justified. This experiment is a good indication of how we can take our big model and retrain it on a smaller dataset for a new arena.

Ablation Study for Soccer: In Table 1 we present the mean IOU score of soccer test images based on employing different potentials in our energy function. For each set of features, we perform 6-fold cross validation to choose the best value of $C \in \{2^{-4}, 2^{-3}, \ldots, 2^{3}\}$ that maximizes the mean IOU across different folds.

We make three observations. First, the inclusion of the field surface potential does not help much. This could be

due to the fact that it does not contain any geometric cues for localizing the field. Second, inclusion of the circle potential increases the IOU by 0.05. Third, we note that sharing the weights between all the lines, and also between all the three circles has as good of a performance as not sharing the weights. This suggests a simpler model does the job.

Hockey Model: For hockey, we trained a model with four weights corresponding to the field surface and a unique weight for the middle circles, the face-off circles, the face-off spots, the corner quarter circles, the vertical lines and the upper sidelines. Thus we have 10 learnable weights. We chose $C = 1$ in our hockey experiments. We achieve a mean IOU of 0.82 on the large dataset and 0.81 on the smaller dataset.

Comparison of Our Method to two Baselines: There is currently no baseline in the literature for fully automatic field localization. All the other methods are semi automatic and rely on methods such as keyframe annotation and camera calibration and as such are different in spirit and not comparable to our method. We hope that by releasing the dataset more baselines can be established. In this work, we derive two baselines based on our segmentation method. As the first baseline, for each test image we retrieve its nearest neighbour (NN) image from the training set based on the field surface segmentation IOU and apply the homography of the training image on the test image.

For the second baseline, we retrieve the nearest neighbour based on the distance transform [17] of the line and circle features obtained from the semantic segmentation network. The results are shown in Table 2.

Semantic Segmentation: We achieve mean IOUs of 0.65 and 0.6 across all classes for the soccer and hockey datasets respectively.

Speed and Number of Iterations. Our method is fast. In Table 3 we present the mean speed and number of iterations for each sport clocked on one core of Intel Xeon 5160 3GHz. We also highlight the total number of states based on the grid size. Note that by using branch and bound we find the exact solution in orders of magnitude less iterations than going over all the states.

Grid Discretization: Our method depends on creating a non-orthogonal grid from rays emanating from each vanishing point. Our grid has to be dense enough so that the important lines in the image fall on the grid. To asses our discretized grid, we take the ground truth vanishing points of the test images in the large hockey dataset and construct a grid for each image. Then, we consider the negative of our loss augmented loss of section 4.4 as a potential and perform branch and bound inference. Ideally since we have assumed perfect vanishing points, if our grid is perfect we should obtain 100 percent mean IOU. However, we are a bit short with a mean IOU of 0.99.

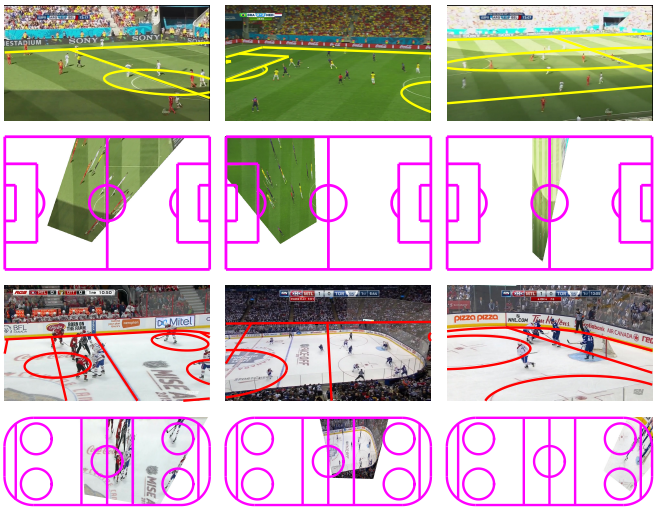


Figure 8: Examples of failure cases

Effect of Vanishing Points Estimation: For the large hockey dataset, we performed an experiment in which instead of estimating the vanishing points, we took the ground truth vanishing point for each image and computed the grid and all the other features as usual. We obtained a mean IOU of 0.9 in contrast to our 0.82 and the baselines of 0.81 and 0.8. This suggest that we can improve our method by getting better vanishing points.

Qualitative Results: In Fig. 7 we project the model on a few test images using the homography obtained with our best features (G+L+C) for soccer and our full set of potentials on hockey. We also project the image on the model of the field. We observe great agreement between the image and the model.

Failure Modes: Fig. 8 shows some failure modes. One main reason for the failure modes is that circle pixels might be classified incorrectly. The other is due to the sensitivity on vanishing points. However, we believe that using temporal information can help overcome these issues.

8. Conclusion and Future Work

In this paper, we presented a new framework for fast and automatic sports field localization. We framed this problem as a deep semantic segmentation task that is fed into a branch and bound method for a fast and exact inference in a Markov Random Field. We evaluated our method on collection of broadcast images from 20 soccer games from World Cup 2014 and eight NHL Hockey matches. We do not take into account temporal information in our energy function. For future work, we intend to construct temporal potential functions and evaluate our method on video sequences. Finally, we aim to extend our method to other team sports such as basketball, rugby and American Football.

References

[1] E. Dubrofsky and R. J. Woodham. Combining line and point correspondences for homography estimation. In *Advances in Visual Computing*. 2008. 1, 2

[2] D. Farin, S. Krabbe, W. Effelsberg, and Others. Robust camera calibration for sport videos using court models. In *Electronic Imaging 2004*, 2003. 1, 2

[3] A. Fitzgibbon, M. Pilu, and R. B. Fisher. Direct least square fitting of ellipses. *PAMI*, 1999. 6

[4] A. Franks, A. Miller, L. Bornn, K. Goldsberry, et al. Characterizing the spatial structure of defensive skill in professional basketball. *The Annals of Applied Statistics*, 2015. 1

[5] X. Gao, Z. Niu, D. Tao, and X. Li. Non-goal scene analysis for soccer video. *Neurocomputing*, 2011. 1

[6] S. Gedikli, J. Bandouch, N. V. Hoyningen-Huene, B. Kirchlechner, and M. Beetz. An adaptive vision system for tracking soccer players from variable camera settings. In *ICVS*, 2007. 2

[7] A. Gupta, J. J. Little, and R. J. Woodham. Using line and ellipse features for rectification of broadcast hockey video. In *CRV*, 2011. 1, 2

[8] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, second edition, 2004. 3

[9] J.-B. Hayet and J. Piater. On-line rectification of sport sequences with moving cameras. In *MICAI*. 2007. 2

[10] J.-B. Hayet, J. Piater, and J. Verly. Robust incremental rectification of sports video sequences. In *BMVC*, 2004. 2

[11] V. Hedau, D. Hoiem, and D. Forsyth. Recovering the spatial layout of cluttered rooms. In *ICCV*, 2009. 6

[12] R. Hess and A. Fern. Improved video registration using non-distinctive local image features. In *CVPR*, 2007. 1, 2

[13] H. K. H. Kim and K. S. H. K. S. Hong. Soccer video mosaicking using self-calibration and line tracking. In *ICPR*, 2000. 1, 2

[14] C. H. Lampert, M. B. Blaschko, and T. Hofmann. Efficient subwindow search: A branch and bound framework for object localization. *PAMI*, 2009. 4

[15] Y. Liu, D. Liang, Q. Huang, and W. Gao. Extracting 3d information from broadcast soccer video. *Image and Vision Computing*, 2006. 1

[16] W.-L. Lu, J.-A. Ting, J. J. Little, and K. P. Murphy. Learning to track and identify players from broadcast sports videos. *PAMI*, 2013. 1

[17] A. Meijster, J. B. Roerdink, and W. H. Hesselink. A general algorithm for computing distance transforms in linear time. In *Mathematical Morphology and its applications to image and signal processing*. 2002. 6, 8

[18] Z. Niu, X. Gao, and Q. Tian. Tactic analysis based on real-world ball trajectory in soccer video. *Pattern Recognition*, 2012. 1

[19] K. Okuma, J. J. Little, and D. G. Lowe. Automatic rectification of long image sequences. In *ACV*, 2004. 1, 2

[20] K. Okuma, D. G. Lowe, and J. J. Little. Self-learning for player localization in sports video. *arXiv preprint arXiv:1307.7198*, 2013. 1

[21] K. Okuma, A. Taleghani, N. De Freitas, J. J. Little, and D. G. Lowe. A boosted particle filter: Multitarget detection and tracking. In *ECCV*. 2004. 1

[22] V. Ramanathan, J. Huang, S. Abu-El-Haija, A. Gorban, K. Murphy, and L. Fei-Fei. Detecting events and key actors in multi-person videos. *arXiv preprint arXiv:1511.02917*, 2015. 1

[23] A. Schwing, S. Fidler, M. Pollefeys, and R. Urtasun. Box in the box: Joint 3d layout and object reasoning from single images. In *ICCV*, 2013. 5

[24] A. G. Schwing, T. Hazan, M. Pollefeys, and R. Urtasun. Efficient structured prediction for 3d indoor scene understanding. In *CVPR*, pages 2815–2822, 2012. 4, 5

[25] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014. 6

[26] T. Tieleman and H. G. Lecture 6.5 - rmsprop, coursera: Neural networks for machine learning. technical report, 2012. 6

[27] X. Tong, J. Liu, T. Wang, and Y. Zhang. Automatic player labeling, tracking and field registration and trajectory mapping in broadcast soccer video. *TIST*, 2011. 1

[28] I. Tsochantaridis, T. Joachims, T. Hofmann, and Y. Altun. Large margin methods for structured and interdependent output variables. In *Journal of Machine Learning Research*, pages 1453–1484, 2005. 1, 3, 5

[29] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *CVPR*, 2001. 5

[30] F. Wang, L. Sun, B. Yang, and S. Yang. Fast arc detection algorithm for play field registration in soccer video mining. In *SMC*, 2006. 1, 2

[31] T. Watanabe, M. Haseyama, and H. Kitajima. A soccer field tracking method with wire frame model from TV images. In *ICIP*, 2004. 1, 2

[32] A. Yamada, Y. Shirai, and J. Miura. Tracking players and a ball in video image sequence and estimating camera parameters for 3D interpretation of soccer games. In *ICPR*, 2002. 1, 2

[33] F. Yu and V. Koltun. Multi-scale context aggregation by dilated convolutions. In *ICLR*, 2016. 6