Automated Top View Registration of Broadcast Football Videos

Rahul Anand Sharma* Bharath Bhat* Vineet Gandhi C. V. Jawahar
KCIS, Center for Visual Information Technology, IIIT-Hyderabad
{rahul.anand@research.,bharath.bhat@research.,vgandhi@,jawahar@}iiit.ac.in

Abstract

In this paper, we propose a fully automatic method to register football broadcast video frames on the static top view model of the playing surface. Automatic registration has been difficult due to the difficulty of finding sufficient point correspondences. We investigate an alternate approach exploiting the edge information from the line markings on the field. We formulate the registration problem as a nearest neighbour search over a synthetically generated dictionary of edge map and homography pairs. The synthetic dictionary generation allows us to exhaustively cover a wide variety of camera angles and positions and reduces this problem to a minimal per-frame edge map matching problem. We show that the per-frame results can be further improved in videos using an optimization framework for temporal camera stabilization. We demonstrate the efficacy of our approach by presenting extensive results on a dataset collected from matches of the football World Cup 2014 and show significant improvement over the current state of the art.

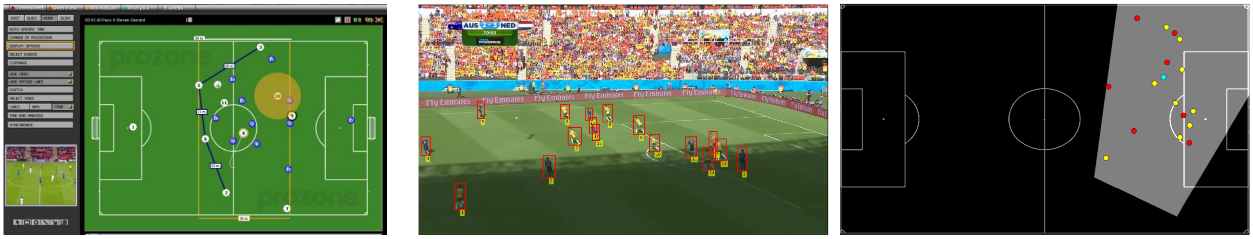
1. Introduction

Advent of tracking systems by companies like Prozone [1] and Tracab [2] has revolutionized the area of football analytics. Such systems stitch the feed from six to ten elevated cameras to record the entire football field, which is then manually labelled with player positions and identity to obtain the top view data over a static model as shown in Figure 1. Majority of recent research efforts [14, 24, 23, 7] and commercial systems for football analytics have been based on such top view data (according to prozone website, more than 350 professional clubs now use their system). There are three major issues with such commercial tracking systems and associated data. First, it is highly labour and time intensive to collect such a data. Second, it is not freely available and has a large price associated with it. Third, such data can not be obtained for analyzing matches where the customized camera installations were not used. It is also difficult for most research groups to collect their own data due to the challenges of installing and maintaining such systems and the need of specific collaborations with the clubs/stadiums.

All the above problems can be addressed, if we can obtain such data using the readily available broadcast videos. However, this is a non trivial task since the available broadcast videos are already edited and only show the match from a particular viewpoint/angle at a given time. Hence, obtaining the top view data first requires the registration of the given viewpoint with the static model of the playing surface. This registration problem is challenging because of the movement of players and the camera; zoom variations; textureless field; symmetries and highly similar regions etc. Due to these reasons, this problem has interested several computer vision researchers [26, 16, 22, 17], however most of the existing solutions are based on computation of point correspondences or/and require some form of manual initialization. Not just that the manual initialization for each video sequence is an impractical task (as shot changes occur quite frequently), such approaches are also not applicable in the presented scenario due to absence of good point correspondences (the football playing surface is almost textureless in contrast to the cases like American football [16]).

Motivated by the above reasons, we take an alternate approach based on edge based features and formulate the problem as a nearest neighbour search to the closest edge map in a precomputed dictionary with known projective transforms. Since, manual labelling of a sufficiently large dictionary of edge maps with known correspondences is an extremely difficult and tedious task, we employ a semi supervised approach, where a large dictionary of 'camera-view edge maps to projective transform pairs' are simulated from a small set of manually annotated examples (the process is illustrated in Figure 3). The simulated dictionary generation allows us to cover edge maps corresponding to various degrees of movement of the camera from different viewpoints (which is an infeasible task manually). More importantly, this idea reduces the accurate homography estimation problem to a minimal dictionary search using the

*Both authors contributed equally



(a)

(b)

Figure 1. (a) A snapshot from prozone tracking system. (b) An example result from the proposed method, which takes as input a broadcast image and outputs its registration over the static top view model with the corresponding player positions. The yellow, red and cyan circles denote the players from different teams and referee respectively.

edge based features computed over the query image. The tracking data can then be simply obtained by projecting the player detections performed over broadcast video frames, using the same projective transform. An example of our approach over a frame from Australia vs Netherlands world cup match is illustrated in Figure 1.

Since the camera follows most of the relevant events happening in the game, it can be fairly assumed that the partial tracking data (only considering the players visible in the current camera view) obtained using the proposed approach is applicable to most of the work on football play style analytics [14]. Furthermore, the knowledge of camera position and movement can work as an additional cue for applications like summarization and event detection (goals. corners etc.), as the camera movement and editing is highly correlated with the events happening in the game. It is also useful for content retrieval applications, for instance it can allow queries like "give me all the counter attack shots" or "give me all the events occurring on the top left corner" etc. The proposed approach can also be beneficial in several other interesting research topics like motion fields for predicting the evolution of the game [19], social saliency for optimized camera selection [28] or automated commentary generation [4].

More formally this work makes following contributions:

1. We propose a novel framework to obtain the registration of football broadcast videos with a static model. We demonstrate that the proposed nearest neighbour search based approach makes it possible to robustly compute the homography in challenging cases, where even manually labelling the minimum four point based correspondences is difficult.

2. We propose a semi-supervised approach to synthetically generate a dictionary of 'camera-view to projective transform pairs' and present a novel dataset with over a hundred thousand pairs.

3. We propose a mechanism to further enhance the results on video sequences using a Markov Random Field (MRF) optimization and a convex optimization framework for removing camera jitter .

4. We present extensive qualitative and quantitative results on two datasets and show significant improvement over the state of the art (using considerably less manually labelled training data)

The proceeding section briefly explains the related work. The semi-supervised dictionary learning approach is described in Section 3.1, followed by the explanation of the proposed matching algorithms. Section 3.3 covers the optimization techniques followed by the experimental results and concluding discussion.

2. Related work

Top view data for sports analytics has been extensively used in previous works. Bialkowski et al. [6] uses 8 fixed high-definition (HD) cameras to detect the players in field hockey matches. They demonstrated that event recognition (goal, penalty corner etc.) can be performed robustly even with noisy player tracks. Lucey et al. [23] used the same setup to highlight that a role based assignment of players can eliminate the need of actual player identities in several applications. In basketball, a fixed set of six small cameras are now used for player tracking as a standard in all NBA matches, and the data has been used for extensive analytics [11]. Football certainly has gained the most attention [14] and the commercially available data has been utilized for variety of applications from estimating the likelihood of a shot to be a goal [24] or to learn a team's defensive weaknesses and strengths [7].

The idea of obtaining top view data from broadcast videos has also been explored in previous works, Okuma et al. [26] used KLT [27] tracks on manually annotated interest points (with known correspondences) and used them in RANSAC [10] based approach to obtain the homographies in presence of camera pan/tilt/zoom in NHL hockey games. Gupta et al. [15] showed improvement over this work by using SIFT features [21] augmented with line and ellipse information. Similar idea of manually annotating initial frame

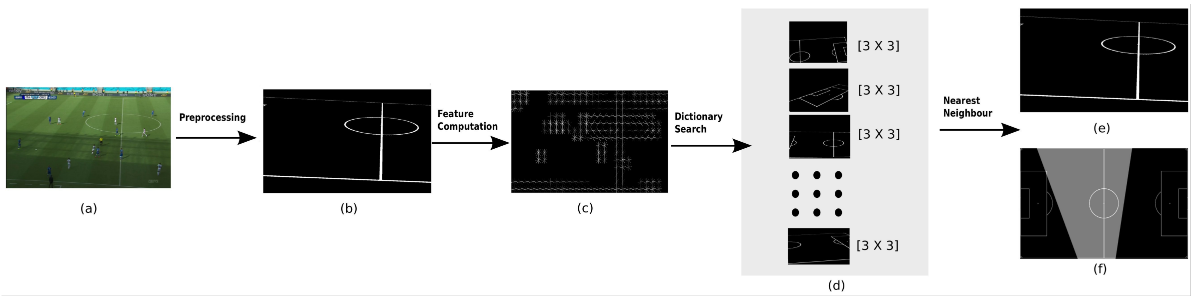


Figure 2. Overview of the proposed approach. The input to the system is a broadcast image (a) and the output is the registration over the static model (f). The image (e) shows the corresponding nearest neighbour edge map from the synthetic dictionary.

and then propagating the matches has also been explored in [22]. Li and Chellapa [20] projected player tracking data from small broadcast clips of American football in top view form to segment group motion patterns. The homographies in their work were also obtained using manually annotated landmarks.

Hess and Fern [16] build upon [26] to eliminate the need of manual initialization of correspondences and proposed an automated method based on SIFT correspondences. Although their approach proposes an improved matching procedure, it may not apply in case of normal football games due to lack of ground visual features. Due to this reason, instead of relying on interest point matches, we move to a more robust edge based approach. Moreover, we use stroke width transforms(SWT) [9] instead of usual edge detectors for filtering out the desired edges. Another drawback of the work in [16] is that the static reference image in their case is manually created, and the process needs to be repeated for each match again. On the other hand, our method is applicable in more generic scenario and we have tested it on data from 16 different matches. The work by Agarwal et al. [3] posed the camera transformation prediction between pair of images as a classification problem by binning possible camera movements, assuming that there is a reasonable overlap between the two input images. However, such an approach is not feasible for predicting exact projective transforms. More recently, Homayounfar et. al [17] presented an algorithm for soccer registration from a single image as a MRF minimization. Their approach relies on vanishing point estimation, which is highly unreliable (in difficult viewpoints, sparse edge detections and shadows). On the other hand, our method is robust is term of preprocessing, just using the pix2pix conditional adversarial network [18] and also extends the automation to video sequences. We further propose a more diverse testing dataset and a more compact training set.

Our work is also related to camera stabilization method of Grundmann et al. [13] which demonstrates that the stabilized camera motion can be represented as combination of distinct constant, linear and parabolic segments. We extend their idea for smoothing the computed homographies over a video. We also benefit from the work of Muja and Lowe [25] for efficient nearest neighbour search.

3. Method

The aim of our method is to register a video sequence with a predefined top view static model. The overall framework of our approach is illustrated in Figure 2. The input image is first pre-processed to remove undesired areas such as crowd and extract visible field lines and obtain a binary edge map. The computed features over this edge map are then used for k-NN search in a pre-built dictionary of images with synthetic edge maps and corresponding homographies. Two different stages of smoothing are then performed to improve the video results. We now describe, each of these steps with detail:

3.1. Semi supervised dictionary generation

Two images of the same planar surface in space are related by a homography $(\mathbf{H})$. In our case, this relates a given arbitrary image from the football broadcast to the static model of the playing surface. Given a point $x = (u, v, 1)$ in one image and the corresponding point $x' = (u', v', 1)$, the homography is a 3×3 matrix, which relates these pixel coordinates $x' = \mathbf{H}x$. The homography matrix has eight degrees of freedom and can ideally be estimated using 4 pairs of perfect correspondences (giving eight equations). In practice, it is estimated using a RANSAC based approach on a large number of partially noisy point correspondences.

However, finding a sufficient set of suitable non-collinear candidate point correspondences is difficult in the case of football fields. And manual labelling each frame is not just tedious, it is also challenging task in several images. Due to these reasons, we take an alternate approach: we first hand label the four correspondences in small set of images (where it can be done accurately) and then use them to simulate a large dictionary of 'field line images (synthetic edge maps) and related homography pairs'. An example of the process is illustrated in Figure 3. Given a training image (Figure 3(a)), we manually label four points

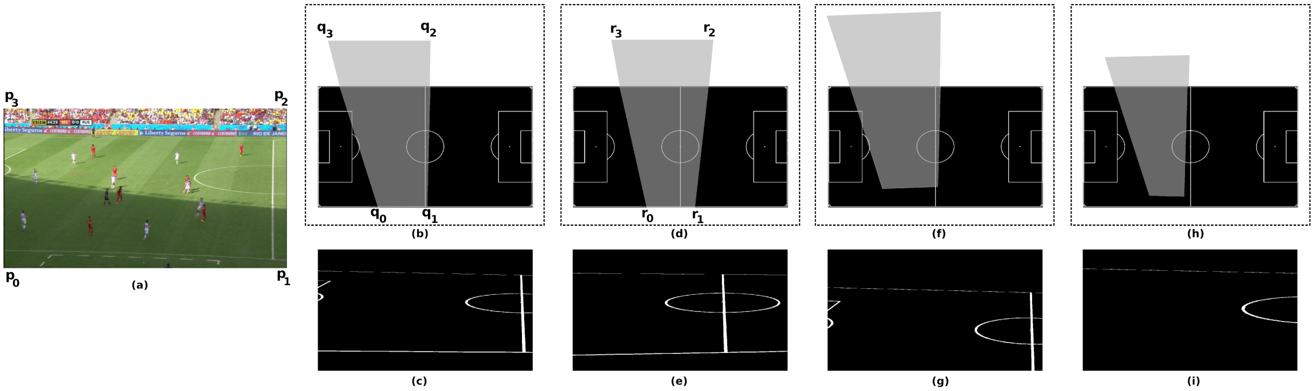


Figure 3. Illustration of synthetic dictionary generation. First column shows the input image and second column shows the corresponding registration obtained using manual annotations of point correspondences. The pan, tilt and zoom simulation process is illustrated in third, fourth and fifth column respectively.

to compute homography $(\mathbf{H_1})$ and register it with the static top view of the ground (Figure 3(b)). We can observe that after applying homography to entire image and warping, the boundary coordinates (p_0, p_1, p_2, p_3) gets projected to points (q_0, q_1, q_2, q_3) respectively. We can now use this to obtain the simulated field edge map (Figure 3(c)) by applying $(\mathbf{H_1^{-1}})$ on the static model (top view). This simulated edge map paired with $\mathbf{H_1}$ forms an entry in the dictionary.

We simulate pan by rotating the quadrilateral (q_0, q_1, q_2, q_3) around the point of convergence of lines q_0q_3 and q_1q_2 to obtain the modified quadrilateral (r_0, r_1, r_2, r_3), as illustrated in Figure 3(d). Using (r_0, r_1, r_2, r_3) and (p_0, p_1, p_2, p_3) as respective point correspondences, we can compute the inverse transform $(\mathbf{H_2^{-1}})$ to obtain Figure 3(e). This simulated image along with $\mathbf{H_2}$ forms another entry in the dictionary. Similarly, we simulate tilt by moving the points q_0q_3 and q_1q_2 along their respective directions and we simulate zoom by expanding (zoom out) or shinking (zoom-in) the quadrilateral about its center. Now, by using different permutations of pan, tilt and zoom over a set of manually annotated images, we learn a large dictionary $D = \{I_j, H_j\}$ where I_j is the simulated edge map, H_j is corresponding homography and $j \in [0 : N-1]$ (we use $N \approx 100K$). We select these images from a larger set of manually annotated images, using a weighted sampling from a hierarchical cluster (using the H matrix as feature for clustering). The permutations of pan, tilt, zoom were chosen carefully to comprehensively cover the different field of views. We can observe that the proposed algorithm is able to generate viewpoint homography pair like Figure 3(i)), which may be infeasible to get using manual annotation (due to lack of distinctive points).

3.2. Nearest neighbour search algorithms

We pose the homography estimation problem as the nearest neighbour search over the synthetic edge map dictionary. Given a preprocessed input image and its edge map x, we find the best matching edge map I_j (or k best matching edge maps) from the dictionary and output the corresponding homography H_j (or set of k homographies). In this section, we present two different approaches we analyzed for computing the nearest neighbours. We specifically choose a direct contour matching approach (chamfer matching) and an image gradient based approach (HOG).

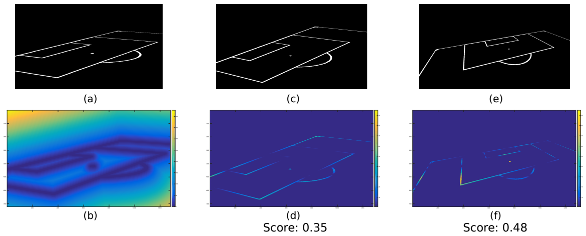


Figure 4. Illustration of chamfer matching. The first column shows the input image x and its distance transform $T(x)$. The second and third column show two different edge maps and their multiplication with $T(x)$. We can observe that image (c) is a closer match and gives a lower chamfer distance.

3.2.1 Chamfer matching based approach

The first method we propose is based on chamfer matching [5], which is a popular technique to find the best alignment between two edge maps. Although proposed decades ago, it remains a preferred method for several reasons like speed and accuracy, as discussed in [29]. Given two edge maps x and I_j, the chamfer distance quantifies the matching between them. The chamfer distance is the mean of the distances between each edge pixel in x and its closest edge pixel in I_j. It can be efficiently computed using the distance transform function $T(.)$, which takes a binary edge image as input and assigns to each pixel in the image the distance to its nearest edge pixel. The chamfer matching then reduces to a simple multiplication of the distance transform on one image with the other binary edge image. The process is illustrated in Figure 4. We use the chamfer distance for the nearest neighbour search. Given an input image x and its distance transform $T(x)$ we search for index j^* in the dictionary, such that

$$j^* = \underset{j}{\text{argmin}} \frac{T(x).I_j}{\|I_j\|_1} \ , \tag{1}$$

where $\|\|_1$ is the ℓ_1 norm and the index j^* gives the index of the true nearest neighbour. Given an epsilon $\epsilon > 0$, the approximate nearest neighbours are given by list of indices j, such that $\frac{T(x).I_j}{\|I_j\|_1} \leq (1+\epsilon)\frac{T(x).I_{j^*}}{\|I_{j^*}\|_1}$.

3.2.2 HOG based approach

The second method is based on HOG features [8], where the nearest neighbour search is performed using the euclidean distance on the HOG features computed over both the dictionary edge maps and the input edge map. So, given the input edge map x and its corresponding HOG features $\phi_h(x)$ we search for j^* in the dictionary, such that

$$j^* = \underset{j}{\text{argmin}} \|\phi_h(x) - \phi_h(I_j)\|_2 \ , \tag{2}$$

where $\|\|_2$ is the ℓ_2 norm.

3.3. Smoothing and Stabilization

For a given input video sequence, we compute k homography candidates independently for each frame using the nearest neighbour search algorithms described above. Just taking the true nearest neighbour for each frame independently may not always be temporally coherent. To remove outliers and to obtain a jerk free and stabilized camera projections, we use two different optimization stages. The first stage uses a markov random field (MRF) based optimization, which selects one of the k predicted homographies for each frame to remove the outliers and discontinuities. The second stage further optimizes these discrete choices, to obtain a more smooth and stabilized camera motion.

3.3.1 MRF optimization

The algorithm takes as input the k predicted homographies for each frame with their corresponding nearest neighbour distances and outputs a sequence of $\xi = \{s_t\}$ states $s_t \in [1:k]$, for all frames $t = [1:N]$. It minimizes the following global cost function:

$$E(\xi) = \sum_{t=1}^{N} E_d(s_t) + \sum_{t=2}^{N} E_s(s_{t-1}, s_t). \tag{3}$$

The cost function consists of a data term E_d that measures the evidence of the object state using the nearest neighbour distances and a smoothness term E_s which penalizes sudden changes. The data term and the smoothness term are defined as follows:

$$E_d(s_t) = \log(P(s_t, t)) \ . \tag{4}$$

Here, $P(s_t, t)$ is the nearest neighbour distance for state s_t at frame t. And

$$E_s(s_{t-1}, s_t) = \|H_{s_t} - H_{s_{t-1}}\|_2, \tag{5}$$

is the Euclidean distance between the two (3×3) homography matrices, normalized so that each of the eight parameters lie in a similar range. Finally, we use dynamic programming (DP) to solve the optimization problem presented in Equation 3.

3.3.2 Camera stabilization

The MRF optimization removes the outliers and the large jerks, however a small camera jitter still remains because its output is a discrete selection at each frame. We solve this problem using a solution inspired by the previous work on camera stabilization [13]. The idea is to break the camera trajectory into distinct constant (no camera movement), linear (camera moves with constant velocity) and parabolic (camera moves with constant acceleration or deceleration) segments. We found that this idea also correlates with the camera work by professional cinematographers, who tend to keep the camera constant as much as possible, and when the movement is motivated they constantly accelerate, follow the subject (constant velocity) and then decelerate to static state [30]. The work in [13] shows that this can be formalized as a L1-norm optimization problem.

However the idea of [13] cannot be directly applied in our case, as we can not rely on interest point features for the optimization, because we are already in projected top view space. We parametrize the projected polygon (for example the quadrilateral $q_0q_1q_2q_3$ in Figure 3) using six parameters, the center of the camera (cx, cy), the pan angle θ, the zoom angle ϕ and two intercepts $(r1, r2)$ (for near clipping plane and far clipping plane respectively). Given a video of N frames, we formulate the stabilization as convex optimization over the projected plane $P_t = \{cx_t, cy_t, \theta_t, \phi_t, r1_t, r2_t\}$ at each frame $t \in [0 : N-1]$. We solve for P_t^* which minimizes the following energy function:

$$\begin{aligned} E_c = \sum_{t=1}^{N}(P_t^* - P_t)^2 + \lambda_1 \sum_{t=1}^{N-1} \|P_{t+1}^* - P_t^*\|_1 \\ +\lambda_2 \sum_{t=1}^{N-2} \|P_{t+2}^* - 2P_{t+1}^* + P_t^*\|_1 \\ +\lambda_3 \sum_{t=1}^{N-3} \|P_{t+3}^* - 3P_{t+2}^* + 3P_{t+1}^* - P_t^*\|_1. \end{aligned} \tag{6}$$

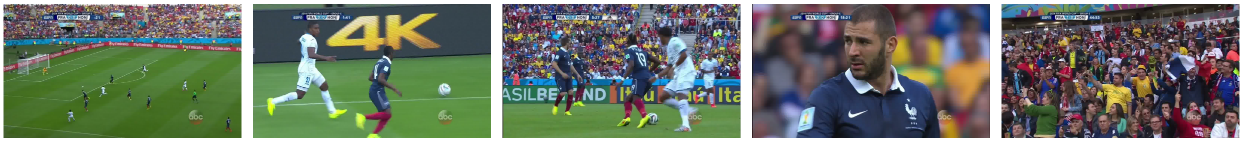


Figure 5. We classify the camera viewpoints from a usual football broadcast into five different categories namely (from left to right) top zoom-out, top zoom-in, ground zoom-out, ground zoom-in and miscellaneous (covering mainly the crowd view).

Accuracy comparison		**Run time comparison**	
	Mean		Mean Time (s)
DSM	83	DSM	0.44
Ours	91.4	Ours	0.21

Table 1. Comparison of our algorithm (using HOG features for matching) with DSM. Both training and testing was done using their soccer dataset [17].

DSM dataset for training			**Our dataset for training**		
	Mean	Median		Mean	Median
DSM test data	91.4	92.7	DSM test data	88.8	90.8
Our test data	76.3	85.6	Our test data	88.4	90.6

Table 2. Results of our algorithm (using HOG features for matching) trained and tested on the two different datasets.

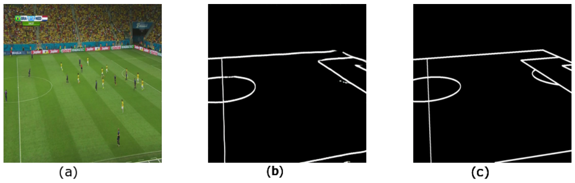
The energy function E_c comprises of a data term and three L1-norm terms over the first order, second order and the third order derivatives and λ_1, λ_2 and λ_3 are parameters. As E_c is convex, it can be efficiently solved using any off the shelf solver, we use cvx [12].

4. Experimental Results

We compare our results with the recent state of the art [17] on their dataset (209 training images and 186 test images). To further demonstrate the efficacy of our approach, we perform experiments on a more diverse testing dataset taken from the matches of entire FIFA 2014 world cup (80 training images and 300 testing images). The diversity is in both the viewpoints and the playing conditions (grass color, lighting etc.). We further show results on broadcast video sequences, comparing with previous methods [26, 22], highlighting the benefits of the camera smoothing and stabilization.

4.1. Results over broadcast images

A football broadcast consists of different kind of camera viewpoints (illustrated in Figure 5) and the field lines are only properly visible in the far top zoom-out view (which though covers nearly seventy five percent of the broadcast video frames). Moreover, The nearest neighbour search takes as input the features over edge maps, hence we need to pre-process the RGB images to obtain the edge maps (only containing the field lines). Henceforth, we propose a two stage pre-processing algorithm:



(a) (b) (c)

Figure 6. Illustration of the pre-processing using pix2pix conditional adversarial network . (a) denotes the given RGB image,(b) denotes the output of the adversarial network, and (c) denotes and ground truth segmentation

4.1.1 Pre-processing

The first pre-processing step selects the top zoom-out frames from a given video sequence. We employ the classical Bag of Words (BoW) representation on SIFT features to classify each frame into one of the five classes illustrated in Figure 5. We use a linear SVM to perform per frame classification (taking features from a temporal window of 40 frames centred around it), followed by a temporal smoothing. Even using this simple approach, we achieve an accuracy of 98 percent, for the top zoom-out class label (trained over 45 minutes of video and tested over 45 minutes of video from another match).

Now, given the top zoom-out images from the video, second pre-processing step is to extract the edge map with field lines. For this we use Pix2Pix [18] for image translation where we give the original RGB image as Input image and field line image as the translated image and train a conditional adversarial network to perform the image translation. An example output is illustrated in Figure 6.

4.1.2 Quantitative evaluation

The results over the DSM dataset [17] are presented in table 1. The proposed method significantly improves upon the DSM approach both in term of prediction accuracy and the run time speeds. The accuracy is measured using mean IOU of the projected image on the static model with the ground truth projections and our approach brings an improvement of more than 8% over DSM. Our approach is also significantly faster while testing.

To further analyze the performance of our method, we propose a new dataset covering all matches from FIFA 2014 world cup. Our training set is more compact (only using 80 images) and the testing set is more diverse, including

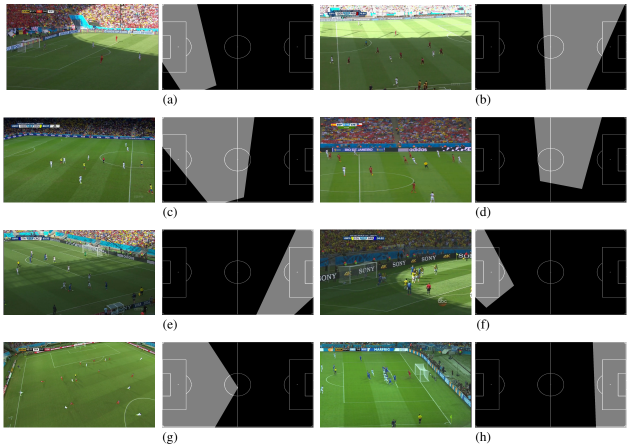


Figure 7. Original images and registered static model pairs computed using the HOG based approach. Covering shadows {(a),(b),(e),(f)}, motion blur {(d)}, varying zoom {(a),(c)}, varying camera viewpoints {(g),(h)}, varying positions {(e),(f)} etc.

varying lighting conditions with prominent shadows, motion blur, varying viewpoints thoroughly covering the playing field. We train two different models, one using DSM training data and one using our training data. We then test their performance on both DSM test data and our test data, We report the results in table 2. The accuracy significantly deteriorate when we use the DSM training data for testing on our data, which highlights the added challenges in our test data. On the other hand, when using our training data, the accuracies are consistent across both the test sets. This demonstrates that the semi supervised dictionary generation is able to cover a significant portion of search space, even when generated from a compact training set of 80 images.

As baseline experiment, we also compared the chamfer matching and HOG matching on our dataset. The HOG based approach outperformed the chamfer matching by a margin of about 6% in mean IOU accuracy and it was significantly faster while testing (due to shorter feature vector and the search speed up using kd trees).

4.1.3 Qualitative evaluation

Results over a small set of images using HOG based approach are shown in Figure 7. We can observe that the predictions are quite accurate over diverse scenarios and the method works perfectly even in cases where manual annotation of point correspondences is challenging in itself (Figure 7(d)). The robustness of our approach over extreme variations in camera angle (Figure 7 (g) and (h)) and challenges like shadows (Figure 7 (a),(b),(e),(f)) and motion blur (Figure 7 (d)) can be observed. The applicability over varying zoom and field coverage is also evident. The reader can refer to the supplementary material for more details, where we provide the results over the entire dataset.

4.2. Results over broadcast videos

Existing video based methods [26, 22] for registration are inapplicable for individual frames as they require user to initialize the homography on the first frame with respect to the model, which is then propogated by tracking points in subsequent frames. This requires manual re-initialization of homography at every shot change, in a typical football video of 45 minutes this happens about 400 times. Clearly our method is superior because it is fully automatic. However, we still perform quantitative comparison with the approach in [26, 22] using two long video sequences by manually labelling 200 frames in them (labelling two frames per second) to compute the mean IOU measure. An approach similar to [26] which has originally been applied on hockey videos based on KLT tracker gave a IOU measure of 35% of the football sequences. This low performance can be attributed to drift and lack of features (on the football field as compared to the hockey rink), due to which the tracking fails after few frames. We implemented a more robust variant using SIFT features instead, which gives a mean IOU measure of 70%. On the other hand, our approach gave a mean IOU measure of 87% when the registration is computed individually on each frame.

MRF evaluation: Using the above mentioned sequences, we then perform MRF optimization by computing k nearest neigbours estimated on the individual frames. We chose k=5 in our experiments. We found that the mean IOU measure improved from 87% to 89% by employing the MRF based optimization over the per frame results.

Convex optimization evaluation: Qualitative results of the camera stabilization are shown in Figure 8 over a video sequence from Chile vs Australia world cup match. The video starts at midfield, pans to left goal post, stays static for few frames and quickly pans back to midfield following a goalkeeper kick. The figure shows the pan angle trajectory of the per frame predictions with and without camera stabilization. We can observe that the optimization clearly removes jitter and replicates a professional cameraman behaviour. The actual and the stabilized video are provided in the supplementary material.

5. Summary

We have presented a method to compute projective transformation between a static model and a broadcast image as a nearest neighbour search and have shown that the presented approach outperforms the current state of the art by about 8% in mean IOU accuracy. Our method can be easily extended to videos without requiring any manual initialization as in previous approaches [26, 22]. Once the dictionary

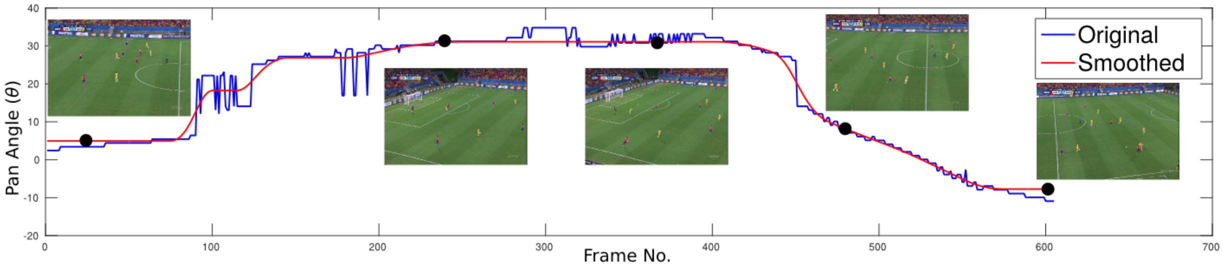


Figure 8. Illustration of stabilization using convex optimization. The blue curve shows the pan angle predicted by the proposed approach on each frame individually. The red curve shows the stabilized pan angle after the convex optimization. We can observe the the smoothed pan angle composes of distinct static, linear and quadratic segments. The black dots denote the frames at respective locations.

is learnt, our method can be directly applied to any standard football broadcast and in fact can be easily extended to any sport where such field lines are available (like basketball, ice hockey etc.). Moreover, the semi supervised dictionary generation allows us to adapt the algorithm even if new camera angles are used in future. The proposed method opens up a window for variety of applications which could be realized using the projected data. One limitation of our approach is that it is only applicable to top zoom-out views and it would be an interesting problem to register other kind of shots (ground zoom-in, top zoom-in shots) using the predictions over top zoom out views, player tracks and other temporal cues.

References

[1] Prozone sports. http://prozonesports.stats.com.

[2] Tracab optical tracking. http://chyronhego.com/sports-data/tracab.

[3] P. Agrawal, J. Carreira, and J. Malik. Learning to see by moving. In *CVPR*, 2015.

[4] E. André, K. Binsted, K. Tanaka-Ishii, S. Luke, G. Herzog, and T. Rist. Three robocup simulation league commentator systems. *AI Magazine*, 21(1):57, 2000.

[5] H. G. Barrow, J. M. Tenenbaum, R. C. Bolles, and H. C. Wolf. Parametric correspondence and chamfer matching: Two new techniques for image matching. Technical report, DTIC Document, 1977.

[6] A. Bialkowski, P. Lucey, P. Carr, S. Denman, I. Matthews, and S. Sridharan. Recognising team activities from noisy data. In *CVPR Workshops*, 2013.

[7] I. Bojinov and L. Bornn. The pressing game: Optimal defensive disruption in soccer. In *Procedings of MIT Sloan Sports Analytics*, 2016.

[8] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, 2005.

[9] B. Epshtein, E. Ofek, and Y. Wexler. Detecting text in natural scenes with stroke width transform. In *CVPR*, 2010.

[10] M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.

[11] A. Franks, A. Miller, L. Bornn, and K. Goldsberry. Counterpoints: Advanced defensive metrics for nba basketball. MIT Sloan Sports Analytics Conference. Boston, MA, 2015.

[12] M. Grant and S. Boyd. CVX: Matlab software for disciplined convex programming, version 2.1. `http://cvxr.com/cvx`, Mar. 2014.

[13] M. Grundmann, V. Kwatra, and I. Essa. Auto-directed video stabilization with robust l1 optimal camera paths. In *CVPR*, 2011.

[14] J. Gudmundsson and M. Horton. Spatio-temporal analysis of team sports–a survey. *arXiv preprint arXiv:1602.06994*, 2016.

[15] A. Gupta, J. J. Little, and R. J. Woodham. Using line and ellipse features for rectification of broadcast hockey video. In *Computer and Robot Vision (CRV), 2011 Canadian Conference on*, pages 32–39. IEEE, 2011.

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

[17] N. Homayounfar, S. Fidler, and R. Urtasun. Sports field localization via deep structured models. In *CVPR*, 2017.

[18] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. In *CVPR*, 2017.

[19] K. Kim, M. Grundmann, A. Shamir, I. Matthews, J. Hodgins, and I. Essa. Motion fields to predict play evolution in dynamic sport scenes. In *CVPR*, 2010.

[20] R. Li and R. Chellappa. Group motion segmentation using a spatio-temporal driving force model. In *CVPR*, 2010.

[21] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, 2004.

[22] W.-L. Lu, J.-A. Ting, J. J. Little, and K. P. Murphy. Learning to track and identify players from broadcast sports videos. *TPAMI*, 35(7):1704–1716, 2013.

[23] P. Lucey, A. Bialkowski, P. Carr, S. Morgan, I. Matthews, and Y. Sheikh. Representing and discovering adversarial team behaviors using player roles. In *CVPR*, 2013.

[24] P. Lucey, A. Bialkowski, M. Monfort, P. Carr, and I. Matthews. quality vs quantity: Improved shot prediction in soccer using strategic features from spatiotemporal data. MIT Sloan Sports Analytics Conference, 2014.

[25] M. Muja and D. G. Lowe. Fast approximate nearest neighbors with automatic algorithm configuration. *VISAPP*, 2009.

[26] K. Okuma, J. J. Little, and D. G. Lowe. Automatic rectification of long image sequences. In *ACCV*, 2004.

[27] J. Shi and C. Tomasi. Good features to track. In *CVPR*, 1994.

[28] H. Soo Park and J. Shi. Social saliency prediction. In *CVPR*, 2015.

[29] A. Thayananthan, B. Stenger, P. H. Torr, and R. Cipolla. Shape context and chamfer matching in cluttered scenes. In *CVPR*, 2003.

[30] R. Thompson and C. Bowen. *Grammar of the Edit*. Focal Press, 2009.