Towards Efficient Ice Surface Localization From Hockey Broadcast Video

by

Pascale B. Walters

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Applied Science
in
Systems Design Engineering

Waterloo, Ontario, Canada, 2021

Author's Declaration

This thesis consists of material all of which I authored or co-authored: see Statement of Contributions included in this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Statement of Contributions

Chapter 3 of this thesis contains content from the previously published paper:

> Pascale Walters, Mehrnaz Fani, David Clausi, and Alexander Wong. 2020. A tool for annotating homographies from hockey broadcast video. In *Journal of Computational Vision and Imaging Systems*, Volume 6, Issue 1.

Mehrnaz Fani, David Clausi, and Alexander Wong contributed to the conceptualization and deployment of this work.

Chapter 4 of this thesis contains content from the previously published paper:

> Pascale Walters, Mehrnaz Fani, David Clausi, and Alexander Wong. 2020. BenderNet and RingerNet: Highly efficient line segmentation deep neural network architectures for ice rink localization. In *Journal of Computational Vision and Imaging Systems*, Volume 6, Issue 1.

Mehrnaz Fani, David Clausi, and Alexander Wong contributed to the conceptualization of this work. Alexander Wong contributed to the editing of a completed draft of this paper.

Abstract

Using computer vision-based technology in ice hockey has recently been embraced as it allows for the automatic collection of analytics. This data would be too expensive and time-consuming to otherwise collect manually. The insights gained from these analytics allow for a more in-depth understanding of the game, which can influence coaching and management decisions. A fundamental component of automatically deriving analytics from hockey broadcast video is ice rink localization. In broadcast video of hockey games, the camera pans, tilts, and zooms to follow the play. To compensate for this motion and get the absolute locations of the players and puck on the ice, an ice rink localization pipeline must find the perspective transform that maps each frame to an overhead view of the rink.

The lack of publicly available datasets makes it difficult to perform research into ice rink localization. A novel annotation tool and dataset are presented, which includes 7,721 frames from National Hockey League game broadcasts.

Since ice rink localization is a component of a full hockey analytics pipeline, it is important that these methods be as efficient as possible to reduce the run time. Small neural networks that reduce inference time while maintaining high accuracy can be used as an intermediate step to perform ice rink localization by segmenting the lines from the playing surface.

Ice rink localization methods tend to infer the camera calibration of each frame in a broadcast sequence individually. This results in perturbations in the output of the pipeline, as there is no consideration of the camera calibrations of the frames before and after in the sequence. One way to reduce the noise in the output is to add a post-processing step after the ice has been localized to smooth the camera parameters and closely simulate the camera's motion. Several methods for extracting the pan, tilt, and zoom from the perspective transform matrix are explored. The camera parameters obtained from the inferred perspective transform can be smoothed to give a visually coherent video output. Deep neural networks have allowed for the development of architectures that can perform several tasks at once. A basis for networks that can regress the ice rink localization parameters and simultaneously smooth them is presented.

This research provides several approaches for improving ice rink localization methods. Specifically, the analytics pipelines can become faster and provide better results visually. This can allow for improved insight into hockey games, which can increase the performance of the hockey team with reduced cost.

Acknowledgements

I would like to thank all the people who made this thesis possible.

Thank you to Prof. David Clausi, Prof. Alex Wong, and Prof. John Zelek for your support and sharing your knowledge with me. To Mehrnaz Fani and the members of the Sports Analytics Research Group: thank you for sharing your ideas and helping me when I got stuck. Thank you to Meghan Chayka and the Stathletes team for believing in a fledgling female sports analytics researcher.

This work would not have been possible without my friends and family. To my parents, Scott and Barbara, and my little sister, Camille. Thank you for all of the love and encouragement. Much gratitude to my family away from home: Leigh Anne, Ben, Kathleen, and Henry. Thank you for taking in this come from away who happened to end up on the Rock. Finally, a big thank you to my friends who helped me hold it together: Lydia, Daniel, Ben, Matthew, Dipika, and Rachel.

Finally, I wish to acknowledge the land on which this thesis was written. Waterloo is located on the traditional territory of the Neutral, Anishinaabeg, and Haudenosaunee peoples; Toronto is located on the traditional land of the Huron-Wendat, the Seneca, and the Mississaugas of the Credit River; and St. John's is located on the ancestral homelands of the Beothuk. I am grateful for the opportunity to have lived and worked on these beautiful lands.

Dedication

To Mommy, Daddy, and Beeble:

> It was the best of times, it was the blurst of times.
>
> —Charles Montgomery Burns

To Sam:

> Being deeply loved by someone gives you strength, while loving someone deeply gives you courage.
>
> —Lao Tzu

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

> Computer vision as a field is an intellectual frontier. Like any frontier, it is exciting and disorganized, and there is often no reliable authority to appeal to. Many useful ideas have no theoretical grounding, and some theories are useless in practice; developed areas are widely scattered, and often one looks completely inaccessible from the other.
>
> —Forsyth and Ponce, 2002 [29]

Computer vision is currently an impressive field of research. It deals with automating tasks that would otherwise be done by the human visual system. There is a wide variety of applications, from medical image analysis to autonomous driving to image retrieval. Computer vision has automated procedures and obtained state-of-the-art performance. This field has expanded significantly in the recent years, in part due to the increased availability and reduction in price of imaging and computation devices, which has spurred the need for new applications and techniques [29].

A new and exciting application of computer vision is in sports. There is a wide range of tasks that need to be tackled within this domain, such as player tracking and event detection [74]. These new insights complement existing systems in sports, such as scouting reports and player development, by automating manual data collection or generating new analytics.

One such sport that can benefit from computer vision is ice hockey. In this dissertation, ice hockey will be the focus of the research and will be referred to by its common name, hockey.

Figure 1.1: Three broadcast frames from one continuous sequence of play. The camera pans, tilts, and zooms to follow the play.

Hockey is a popular sport in Canada, with several professional leagues operating, such as the NHL (National Hockey League) and CHL (Canadian Hockey League). Furthermore, Hockey Canada, the national governing body for hockey in Canada, reports that they had 605,963 players and 92,622 coaches registered for the 2019-2020 season [3].

Video analytics of hockey games can be used to provide teams with an advantage over their competitors, whereby they can gather more data about game events. These data can be used to influence coaching strategies and management decisions. In addition, the data can increase fan engagement as sports consumption becomes more digital [81]. The sports analytics market is rapidly growing and is anticipated to reach revenues of $4.5 billion by 2024 [2].

With many recent developments in the field of computer vision, automatic generation of sports analysis data from video has become possible. Existing computer vision solutions analyze video feed from several cameras placed in calibrated locations throughout the arena [5]. While this technique can be effective, it requires specialized hardware to be deployed at all arenas where games are played. In situations where this may not be possible, analytics derived from broadcast footage is an appropriate substitute. Broadcast footage refers to video that is collected live from at least one camera and distributed for viewing on television or other platforms.

1.1 Motivation

In hockey, analytics can provide teams with advanced statistics about the individual players and the team as a whole. These statistics are then used by the teams to influence coaching decisions, such as assessing player development and preparing to face a certain opponent in an upcoming game. Management decisions can also be improved, especially when scouting

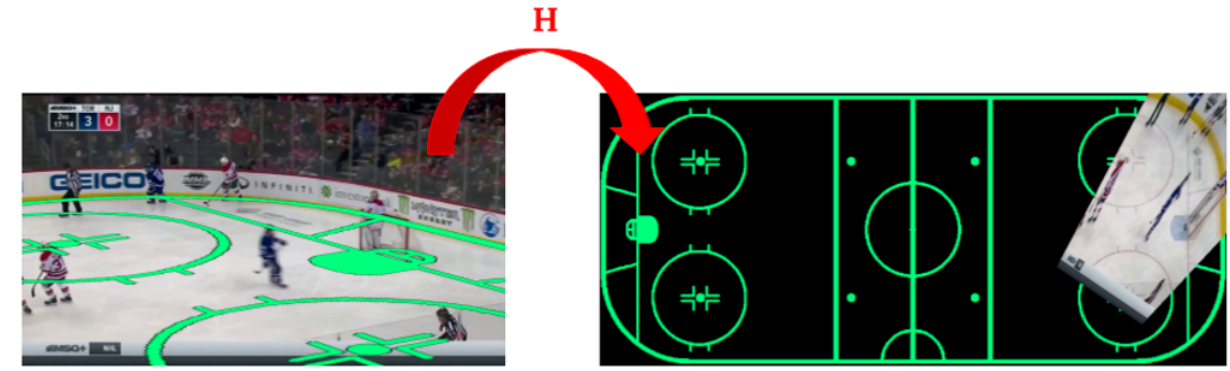


Figure 1.2: Sports field localization involves determining the transform H from the hockey broadcast video frame (left) to the overhead view of the rink (right).

for young players coming up through the minor hockey leagues and other players within the league looking for a trade.

Collecting these data needed to generate analytics can be resource intensive, especially if they are manually annotated, making it time-consuming and expensive. This can be prohibitive for many teams, especially those that may have a lower budget and fewer employee time resources. Extracting hockey analytics from broadcast video makes them easier to obtain, which, in turn, makes the sport more equitable for all. This is especially important as it aligns with the mission of several organizations that aim to increase the availability of hockey, such as the Hockey Diversity Alliance [4], Black Girl Hockey Club [1], and the Professional Women's Hockey Players Association [7].

1.2 Thesis Overview

Analytics from hockey broadcast video are difficult to extract due to the motion of the broadcast camera. The camera operator pans, tilts, and zooms to follow the play. The angular subtense is much lower when viewing a hockey broadcast than if the viewer were there in person. Therefore, the camera operator uses a camera with a long focal length and captures a small area of the ice at a time [66]. The panning, tilting, and zooming of the camera allows the viewer to comfortably observe the game. Fig. 1.1 shows three frames from an NHL broadcast clip where the camera pans, tilts, and zooms to follow the play.

Sports field localization is required to compensate for the motion of the camera to de-

termine the absolute locations of the players, puck, and referees on the ice surface. Fig. 1.2 shows a frame from a hockey broadcast video and its localization. This is a fundamental task in extracting analytics from broadcast video, as once the absolute positions are known, insights about the game can be extracted.

This work attempts to contribute to the research of ice rink localization. A novel annotation tool for collecting ground truth data from hockey broadcast video and a dataset for hockey rink localization are presented. Research methods that aim to solve the problem of hockey broadcast video localization are then discussed. Particularly, this work focuses on techniques that reduce inference time and improve smoothness of the output. First, there are two small and fast methods for segmenting lines on the ice surface. Then, two approaches for smoothing the ice rink localization for a sequence of broadcast video frames are presented. They are methods for extracting camera parameters, and simultaneous sports field localization and smoothing using deep networks. The contributions of this work are new directions of research for improving ice rink localization, which can lead to improved efficiency for automatic analysis of hockey video.

Chapter 2

Background

Due to recent significant progress in the field of computer vision and the increased availability of sports video, there has been much research into automatically extracting sports analytics from video. This chapter reviews computer vision applications in hockey, sports field localization, and general homography estimation techniques.

2.1 Computer Vision in Hockey

As part of a pipeline to automatically extract analytics from hockey video, computer vision techniques have been applied to several tasks. These include:

- Player detection and classification [54, 32],
- Player tracking [55, 82, 65]
- Player identification [16],
- Player pose estimation [61],
- Individual action recognition [14, 27, 80],
- Event detection and classification [78, 76, 55, 75, 25, 82],
- Fight detection [60, 43, 10],
- Broadcast video rectification [33, 18, 51, 13, 83, 40, 26, 63, 77],

- Puck tracking [66, 79], and
- Logo detection [47].

This list is in a rough order of increasing complexity. When automatically extracting analytics from hockey, there first needs to be an understanding of where the players are in each frame (player detection), the team to which they belong (player classification), and who they are (player re-identification). Next, to gain insight into the game from a sequence of frames, player tracking is performed to link player detections across a video. This gives some understanding of how the players move, and it can also be further used to estimate the pose of each player and recognize what the player is doing, such as skating forwards or shooting (individual action recognition).

In the sport of hockey, individuals perform actions and events happen at the game level. For instance, there could be five players skating forwards and three players skating backwards, but the event would be classified as a zone entry. The event is an action that may involve multiple players. Beyond classifying the event, it also needs to be associated with a time (event detection).

Computer vision tasks in hockey require a video source. Many professional hockey games are broadcast for consumption by viewers who are not physically present at the game. Using this broadcast footage for computer vision tasks means no additional specialized hardware is required for generating analytics. Despite broadcast footage being readily available, it does have some specific challenges. The broadcast camera pans, tilts, and zooms to follow the play. Compensating for this camera motion is an additional task in the analytics pipeline.

Some additional areas of research that involve computer vision in hockey are puck tracking and team logo detection. The puck tracking task can predict the location of the play by using the absolute location of the puck on the ice as a surrogate [66]. Locating and identifying team logos and jersey colours can be further used to classify players into teams.

Recent advances in deep neural networks have shown that using them can achieve state-of-the-art results in computer vision tasks, including automatic generation of hockey analytics [79]. Many of the highest performing methods listed in this section use convolutional neural networks (CNNs) or other deep learning architectures [16, 76, 80, 78, 79, 54, 32, 18, 40, 61].

2.2 Sports Field Localization

The sports field localization task involves determining the planar transform between the view of the hockey rink from the broadcast video frame and an overhead view of a hockey rink template. This transform is defined by a homography matrix between two views of the plane of the ice surface (Fig. 1.2).

The homography matrix H is a 3×3 matrix with eight degrees of freedom. It relates the transformation between two planes, up to a scale factor s . Equation 2.1 shows a sample generic homography matrix. It relates the point $\begin{bmatrix} x' & y' & 1 \end{bmatrix}^T$ on the rink model to the point $\begin{bmatrix} x & y & 1 \end{bmatrix}^T$ on the ice surface in the broadcast frame. The elements of H are a combination of translation, rotation, and scale [34].

$$s \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = H \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{2.1}$$

This project focuses on broadcast video of NHL games. There are specific challenges associated with this task. Despite the constant dimensions of the rink in the NHL, there are varying appearances between rinks in the league, such as graphics on the ice surface and the boards, illumination, and broadcast camera location. The broadcaster may also overlay graphics on the video to enhance the viewer experience by adding a score bug, to display the current score of the game and time in the game, or a news feed. Three frames in different rinks with different broadcasters are shown in Fig. 2.1.

Over the course of the game, the broadcast features advertisements, player interviews, and segments with a panel of hockey experts, as well as the game itself. Furthermore, coverage of the hockey game can feature footage from several cameras set up around the arena. Hockey rink localization in this dissertation focuses on footage from the main broadcast camera, which is located in the arena stands above the centre ice line.

The simplest way to calculate the transform between the plane of the ice surface in the broadcast video and the rink model is to detect field markings, such as points, lines, and line intersections in the frame then associate them with the corresponding markings in the model. The homography transform can then be estimated with the direct linear transform (DLT) algorithm [34]. The DLT algorithm does not give an accurate transform matrix in the presence of noise in the two sets of image points. A set of inlier point correspondences, found with the random sample consensus (RANSAC) algorithm, is needed to robustly estimate the homography transform. In hockey camera calibration, this task is non-trivial:

Figure 2.1: Three frames showing differences in appearance of the rinks and broadcasts across the NHL.

the markings are usually small, the field is textureless, and the markings may not be in the frame or occluded by players [63].

2.3 Sports Field Localization Methods

Playing surface localization is an open research problem in many sports. In sports broadcasts, the camera pans, tilts, and zooms to follow the play, so that important events fill the frame [66]. In the literature, there have been several methods described that attempt to compensate for the camera's motion. Some of these works also present datasets for sports field localization.

Early methods tend to use feature detection methods, such as scale-invariant feature transform (SIFT) [53], speeded up robust features (SURF) [11], and scale-invariant feature operator (SFOP) [30]. Okuma *et al.* manually find an initial homography estimate, then use Kanade-Lucas-Tomasi (KLT) features to find homography transforms between frames in broadcast hockey video [65]. Hayet *et al.* perform soccer field localization using colour-based line detection and KLT features [36]. Hess and Fern use the Harris affine region detector and SIFT features for registering broadcast football footage [38]. Hayet and Piater use colour-based line tracking, as well as KLT and Harris features for broadcast soccer video [35]. Gupta *et al.* use SFOP and SIFT features for broadcast hockey video [33]. Wen *et al.* use SURF features and colour-based playing field extraction for basketball [84]. Zeng *et al.* use a similar strategy for field hockey video [87].

These early methods tend to only perform under controlled conditions and don't translate well to sports or situations for which they weren't originally developed. This means that changes in factors such lighting, player appearance, and playing surface appearance could cause these techniques to fail.

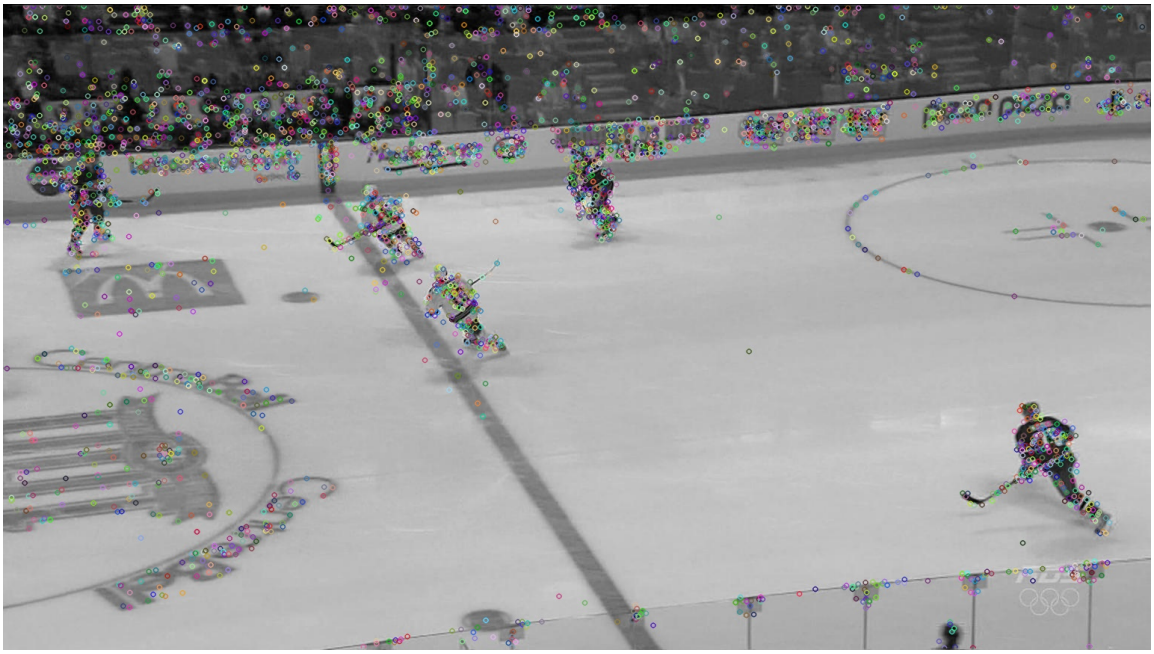


Figure 2.2: SIFT keypoints (coloured circles) for a hockey broadcast video frame. Most of the keypoints are detected on the stands and players, rather than the ice surface.

SIFT keypoints for a hockey broadcast frame are shown in Fig. 2.2. Most of the keypoints are on the players and the spectators in the stands, rather than on the ice surface.

The most common method for sports field localization uses line detection and matching. Some methods described in the previous paragraph use both feature detection and line detection [35, 33]. Thomas uses the Hough transform to extract lines from soccer video and matches them to a field template with a least squares algorithm [73]. Kim and Hong use the Hough transform to detect lines on the playing surface of soccer video and camera parameter-guided line tracking to estimate the camera parameters [46]. Carr *et al.* align the lines visible in field hockey video to a field template with gradient-based optimization [15]. Dubrofsky and Woodham use both point and line correspondences for estimating the homography with a hockey dataset [26]. Yao *et al.* also detect lines on a soccer field with the Hough transform and determine point correspondences with a field template by parameterizing the nature of four line crossings in the frame [86].

Homayounfar *et al.* detect lines with a VGG16 semantic segmentation network, use them to minimize the energy of the vanishing point, and estimate the camera position via

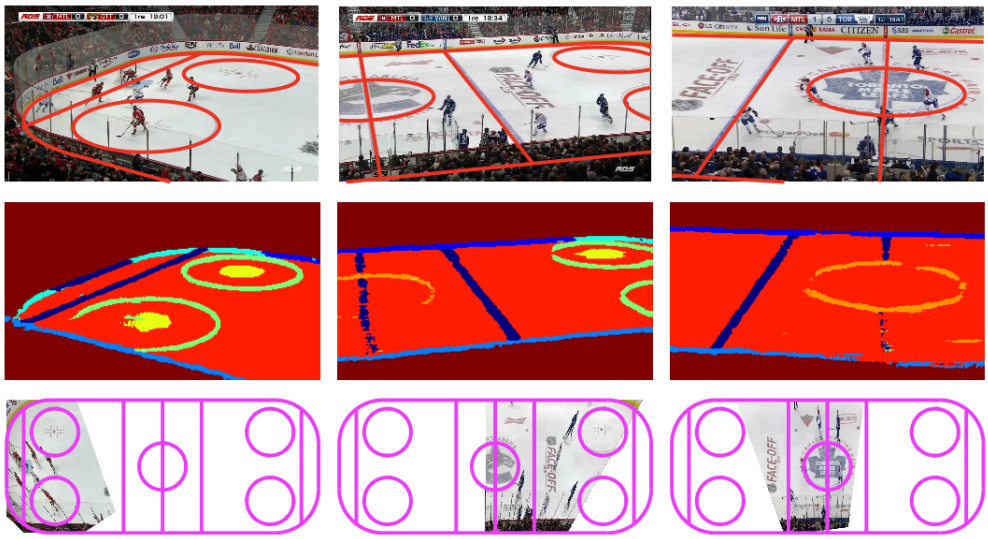


Figure 2.3: Line segmentation in the method proposed by Homayounfar *et al.* [40]. The results of their line segmentation with a VGG16 network are in the middle row.

branch and bound [40]. They perform their work on the World Cup soccer dataset and the SportLogiq hockey dataset. The output of their line segmentation method and sports field localization are shown in Fig. 2.3.

Sharma *et al.* and Chen and Little use the pix2pix network to extract the lines from the playing surface on a dataset of soccer broadcast video. Both works then compare the extracted edge images to a database of synthetic edge images with known homographies in order to localize the playing field [17, 69]. Skinner and Zollman performed playing field localization for rugby games filmed with a smartphone camera from the stands. They extract lines with the Hough transform and match the vertical lines to a template of the pitch [71]. Cuevas *et al.* detect the lines on a soccer field and classify them to match them to a template of the field [22]. Tsurusaki *et al.* use the line segment detector to find intersections of lines on a soccer field, then match them to a template of a standard soccer field using an intersection refinement algorithm [77]. These methods tend to work relatively well, however they struggle in sports where the players are large compared to the playing surface, as in hockey. They also tend to fail when the video frame does not have many lines in it.

Another category of sports field localization techniques extract zones from the playing surface, rather than the lines, as this may be more robust [68]. Zeng *et al.* extract the zones and use feature detection in their method [87]. Sha *et al.* detect the zones from

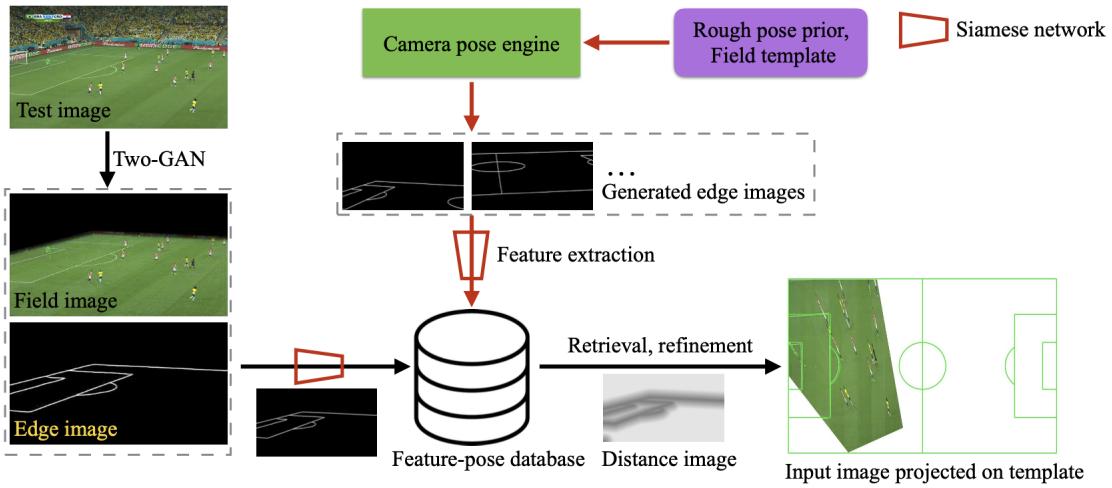


Figure 2.4: Sports field localization pipeline presented by Chen and Little [17]. They generate a feature-pose database to match the input broadcast video frame to a frame with known homography matrix.

soccer and basketball datasets. They initialize the camera pose estimation through a dictionary lookup and refine the pose with a spatial transformer network [68]. Tarashima performs semantic segmentation of the zones as part of a multi-task learning approach for a basketball dataset [72].

Furthermore, there are sports field localization methods that do not necessarily fall into the previously discussed categories. As discussed in the previous two paragraphs, there are some methods that use a dictionary search step in their sports field localization pipeline [69, 17, 71, 22, 68]. The architecture of Chen and Little's sports field localization pipeline that uses a feature-pose database to match the segmented lines from the input broadcast video frame to a frame with known homography matrix. The dictionary search methods require a large training set that represents all the test data the method will see. The database of frames can also become quite large, increasing the inference time.

Ghanem *et al.* perform image patch matching in a football dataset [31]. Citraro *et al.* segment keypoints based on intersections of the lines on the playing surface and match them to a template for basketball, volleyball, and soccer datasets [21]. Similarly, Nie *et al.* segment a uniform grid of points on the playing surface and compute dense features for localizing video of soccer, football, hockey, basketball, and tennis [63]. Jiang *et al.* propose a method for soccer and hockey video to refine homography estimates by concatenating the warped template and frame, then minimizing estimation error [45].

The wide variety of methods that have been described in the literature shows that there is no one method that works particularly well for all sports applications. Furthermore, these methods do not tend to publicly release their code and dataset. It is therefore incredibly difficult to assess the accuracy and inference time for our project requirements. Recently described techniques show that deep network architectures achieve better performance with faster computation [45, 72, 21, 17].

Further research is needed into a method for hockey rink localization that can be evaluated on a novel dataset of NHL broadcast video, described in Chapter 3. Most of the sports field localization methods that are presented in the literature do not report their methods on hockey broadcast video, with the exception of Homayounfar *et al.* [40], Jiang *et al.* [45], and Nie *et al.* [63]. A new method is needed that overcomes the shortcomings of these methods, namely a short inference time, high accuracy, and temporal constancy over a video sequence of hockey broadcast frames.

2.3.1 Evaluation Metrics

Sports field localization methods tend to use either IOU_{part} or $\text{IOU}_{\text{whole}}$ or both. These two metrics rely on comparing the ground truth transform to the inferred transform.

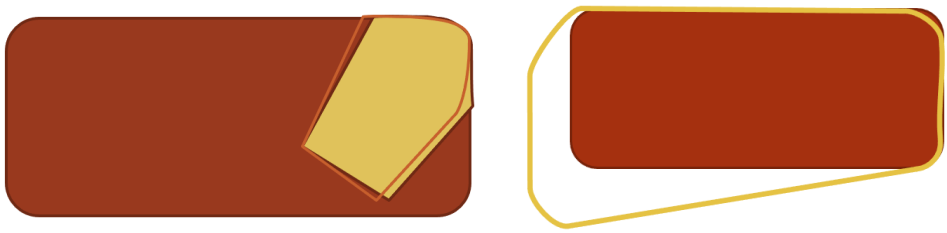
IOU_{part} is the intersection over union of the frame warped onto the rink according to the ground truth and predicted homographies. In Fig. 2.5a, the yellow shape represents the ground truth broadcast frame warp and the shape outlined in orange is the predicted broadcast frame warp. The IOU_{part} would be the intersection over union of these two shapes.

$\text{IOU}_{\text{whole}}$ is the intersection over union of the whole rink warped according to the predicted sports field localization transform. In Fig. 2.5b, $\text{IOU}_{\text{whole}}$ is the intersection over union of the ground truth (red rink shape) and the predicted rink (yellow outline).

It is recommended to only use the $\text{IOU}_{\text{whole}}$ score of a method because it considers the whole playing surface, rather than just the part of the field visible in the broadcast frame [21]. In the worst case scenario, the IOU_{part} could be perfect while the $\text{IOU}_{\text{whole}}$ is much lower, due to it considering the performance of the method on the whole field.

2.4 Sports Field Localization Datasets

Research within the area of sports field localization can be difficult due to the lack of publicly available datasets and the variety of test datasets on which methods in the literature



(a) IOU_{part} (b) $\text{IOU}_{\text{whole}}$

Figure 2.5: Evaluation metrics for ice rink localization.

	IOU_{part}	$\text{IOU}_{\text{whole}}$	Inference Time (s)
Homayounfar *et al.* [40]	83.0		0.44
Sharma *et al.* [69]	91.4		0.21
Chen and Little [17]	94.5	89.4	0.5
Sha *et al.* [68]	93.2	88.3	**0.004**
Citraro *et al.* [21]		**93.9**	0.125
Jiang *et al.* [45]	95.1	89.8	1.36
Nie *et al.* [63]	**95.9**	91.6	0.5
Tsurusaki *et al.*[2] [77]	97.4		

Table 2.1: Performance of sports field localization methods on the soccer World Cup dataset. Numbers in **bold** are the best performance for each metric.

are evaluated. To our knowledge, the only publicly available dataset is the soccer World Cup dataset[1]. The dataset contains broadcast video from 20 soccer games, and has 209 training frames and 186 testing frames [40]. There are eight papers that have reported the performance of their methods on this dataset [40, 69, 17, 68, 21, 45, 63, 77]. The performance of these methods are reported in Table 2.1.

Sample output of three of the sports field localization methods on the soccer World Cup dataset is shown in Fig. 2.6. The method proposed by Sha *et al.* achieves the lowest inference time. Despite including a dictionary search step for camera pose initialization, which could potentially slow the method, they are able to reduce the search space and use an end-to-end architecture [68]. The authors report that their method struggles with the

[1] http://www.cs.toronto.edu/~namdar/data/soccer_data.tar.gz

[2] The performance of the method proposed by Tsurusaki *et al.* is only reported on the 18 best performing frames in the test dataset.

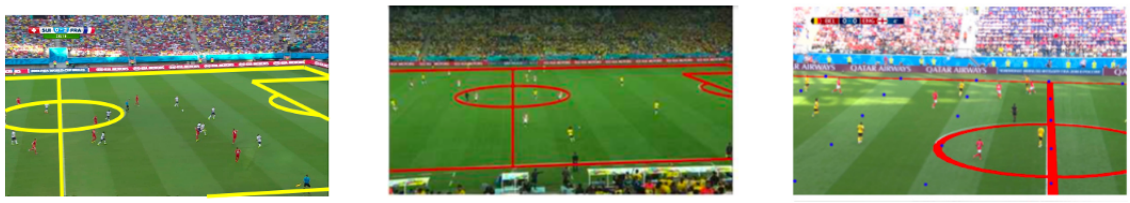


Figure 2.6: Sample output of three sports field localization methods on the soccer World Cup dataset. From left to right: Homayounfar *et al.* [40], Chen and Little [17], Nie *et al.* [63].

semantic segmentation component of their architecture, due to insufficient training data, which gives lower IOU_{part} and $\text{IOU}_{\text{whole}}$ scores. Citraro *et al.* achieves the best $\text{IOU}_{\text{whole}}$ score, which, they argue, is the only accuracy metric that is significant [21]. Their sports field localization pipeline is unique in that it considers the locations of the players on the field, which requires them to augment the World Cup dataset by manually annotating the player positions. When this component is removed, the $\text{IOU}_{\text{whole}}$ drops to 90.5, which is lower than most of the other methods. Nie *et al.* achieve the highest IOU_{part} score using their dense features approach, however their $\text{IOU}_{\text{whole}}$ is still lower than Citraro *et al.* [63].

Other datasets that have been described for sports field localization but may or may not be available publicly are listed in Table 2.2.

2.5 General Homography Estimation

Sports field localization is a special case of homography estimation, where the structure of the playing field plane is known [63]. There are two main approaches for estimating the homography transform between two views of a plane: keypoint detection and parameter regression.

Keypoint detection involves finding salient points on the two views and their correspondences. Some techniques include using descriptors such as SIFT [53], SURF [11], and SFOP [30]. Once corresponding keypoints are found, the algorithm with RANSAC is used to get an estimate of the homography [34]. Deep learning techniques can also be used to detect keypoints in situations without uniform appearance and dynamic scenes [63]. For example, human pose estimation has had success with deep network architectures [50, 20, 56].

Traditional image processing techniques for homography estimation that use hand-crafted features, such as corner and edge detection, tend to face certain downfalls. First,

Dataset	Number of Images	Publicly Available?
World Cup Soccer [69]	500	✗
World Cup Soccer [40]	209 (train), 186 (test)	✓
Hockey [40, 45]	1.67M	✗
Basketball [21]	50,127	?
Volleyball [21]	12,987	?
MLS Soccer [21]	14,160	✗
Basketball [72]	1,232	✗
College Basketball [68]	526 (train), 114 (test)	✗
SportsFields [63]	1,833 (train), 1,134 (test)	✗

Table 2.2: Datasets for sports field localization. Datasets that have a ? in the Publicly Available column were reported in their respective papers as being made available, but do not seem to actually be so.

these feature detectors are sensitive to large variation in appearance and scenes that are not static. Deep networks use learned features, which capture context from a larger area of the image, as compared to the handcrafted feature detectors. Deep networks are more robust to local changes in appearance.

Deep network architectures have also made it possible to directly estimate the parameters of the homography transform between two frames [24, 62, 48]. DeTone were the first to do this task, and they parameterized the homography matrix as the mapping of four corners from one image to the next image in a randomly selected rectangle. They use a 10-layer network on the MS-COCO dataset, which is tested by randomly perturbing the corners of the images [24]. The four point parameterization is shown in Fig. 2.7. Nguyen *et al.* propose a method for stitching together aerial images. They use an unsupervised approach that extends the work of DeTone *et al.* by adding a spatial transform layer and a photometric loss. These two methods work well for image pairs that can be fully aligned with a homography transform [62]. Le *et al.* attempt to solve the problem for dynamic scenes (i.e., those where there is motion in the foreground and background). They train a deep neural network for multi-task learning that performs both homography estimation and dynamic content detection [48].

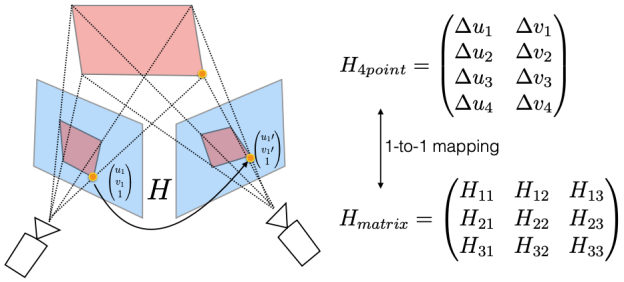


Figure 2.7: Four point parameterization of the homography [24]. H_{matrix} is the 3×3 homography matrix that represents the perspective transform between the two squares. H_{4point} is the 4-point parameterization of the homography. It represents the change in the coordinates of the corners of a randomly selected rectangle from the first image.

2.6 Conclusion

This chapter extensively explores work in the literature for computer vision applications in hockey, sports field localization, and general homography estimation methods. There is a lack of publicly available datasets on which to perform research for ice rink localization for hockey broadcast video. The many approaches that have been published show that no consensus has been reached for the optimal way to solve the problem of sports field localization. Each new contribution to this field proposes a completely new method, rather than attempting to incrementally improve the methods that already exists. Some areas in which these proposed approaches can be enhanced are in reducing inference time and improving visual coherence in the output.

Chapter 3

Annotation Tool and Dataset

Many techniques have been described in the literature to perform sports camera calibration with deep learning methods [40, 69, 17, 72, 45, 21, 68]. However, the authors of these methods do not release the datasets that they have used for training and testing, with the exception of the World Cup dataset for soccer games [40]. Therefore, a novel dataset is required in order to develop new ice rink localization techniques.

In this chapter, a new annotation tool for collecting homographies from frames of hockey broadcast videos is presented. It relies on annotating corresponding points on each frame and a model of the overhead view of the ice surface. With this tool, we have collected a dataset of frames, each of which has a corresponding homography and time in its video sequence.

3.1 Related Work

In the literature, each method for performing sports field localization tends to come with its own sports field localization dataset. These datasets, with the exception of the World Cup dataset [40], have not publicly been made available.

Several sports have been the focus of field localization techniques. Datasets described in the literature include volleyball [21], basketball [72], and hockey [45]. Table 2.2 compares the datasets used for sports field localization methods.

The only publicly available dataset for this problem, World Cup Soccer, has 395 annotated frames. Each frame has an associated homography, but no temporal information (i.e., the frames have been randomly collected from broadcast footage) [68].

All methods report that their datasets have been annotated with point correspondences, and the associated homography determined with the DLT algorithm. To our knowledge, no annotation tool for sports field localization has been released.

Tarashima describes an annotation method for their basketball dataset, whereby only pre-specified intersections of lines on the playing surface and corresponding points on overhead playing surface model are annotated [72]. Citraro *et al.* describe a semi-automated method that was used for their datasets [21]. The annotation tool automatically tracks keypoints and the user provides corrections as needed.

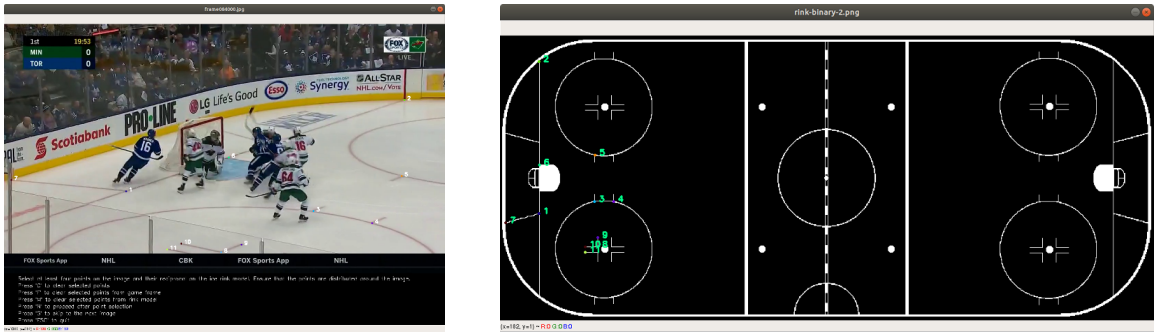
3.2 Motivation

This research is the result of a partnership between Stathletes, a Canadian hockey analytics company, and the Sports Analytics Research Group, within the Vision and Image Processing Lab at the University of Waterloo. Stathletes analysts manually annotate broadcast footage of hockey games to collect analytics that they then distribute to teams and leagues. Their analysts are very skilled and quick at annotation tasks, such as rapidly and accurately clicking on an image with a mouse, and this annotation tool was developed with their expertise in mind.

There were several design constraints that were faced when developing this annotation tool. The Stathletes analysts are not necessarily familiar with coding and using the command-line interface of a computer. This meant that the tool needed to be portable, and not provided to them as a series of code files and dependencies that needed to be installed on their computers. The tool needed to be shared as a single file that contained all code dependencies is opened and run on the user's computer by double-clicking on an icon, similar to other standard Windows applications.

These analysts also work on computers running the Microsoft Windows operating system, rather than the computer on which this code was developed, which runs the Ubuntu operating system. Finally, we only had one graduate student to develop this application. This meant that the tool could not have too many features, which would take too long to develop.

The challenge with this application was striking a balance between having a tool that the Stathletes analysts could use efficiently and not having so many features that the research team was spending too much time on development, rather than performing research.



(a) Points annotated on a frame from a hockey broadcast video.

(b) Corresponding points annotated on the overhead model of an NHL rink.

Figure 3.1: Annotation of points on the frame and rink model using the homography annotation tool.

3.3 Annotation Tool

Obtaining the true homography matrix for each frame in a broadcast video sequence proved to be impossible, as there is no way to obtain the camera parameters from the broadcast. For this reason, we needed to collect our own annotated dataset.

The annotation tool works by having the operator select point correspondences between frames in NHL broadcast footage and a standard model of the rink surface. Point correspondences were selected as the means for collecting the ground truth homography matrices because it was easily implemented in Python with OpenCV's `findHomography` method [6]. This function uses the DLT algorithm with RANSAC to determine the best homography matrix between the corresponding 2D point annotations.

The tool was developed in Python and released as an executable using the PyInstaller library for use by analysts at Stathletes [8]. To use the tool, the user selects corresponding points, alternating between the frame and the rink model (Fig. 3.1). The points are numbered and coloured to match on the frame and rink template.

Users are instructed to select as many points as possible. The best points (i.e., highest precision) are located at the intersections and ends of lines on the playing surface. For example, the intersection of the goal line and boards, the intersection of the hash marks and faceoff circle, and the base of the goal post. The users are also able to zoom in and out to ensure that they are clicking on the correct position.

After the user has selected at least four corresponding points, representing the eight

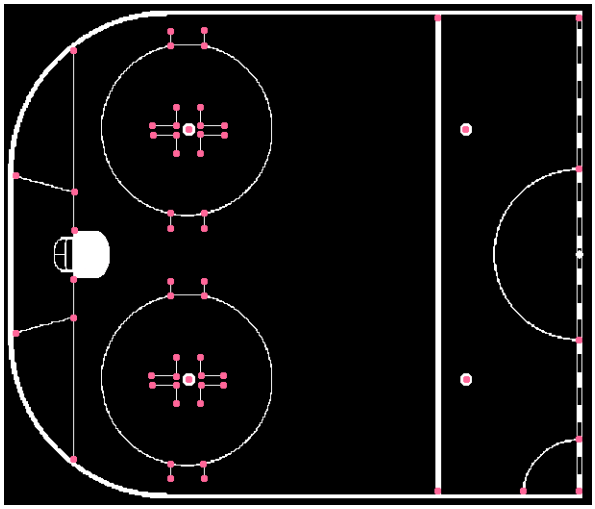


Figure 3.2: The best points to annotate with the ice rink localization annotation tool are at intersections and ends of lines on the ice surface (pink dots).

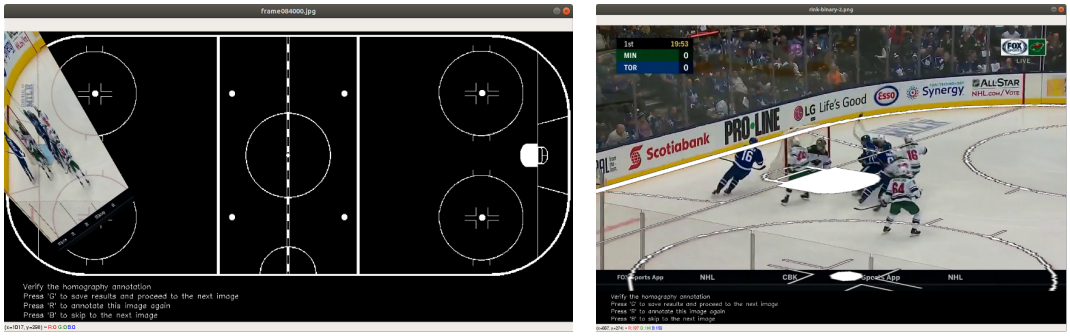
degrees of freedom in the homography matrix, the tool then calculates and displays the results of that homography warping. The frame is warped and overlayed on top of the rink model and the rink model is warped on top of the frame (Fig. 3.3).

The homography is determined with DLT and RANSAC for outlier rejection. We have tuned the RANSAC parameters for the highest visual accuracy, while requiring the fewest point annotations.

The user can then accept, reject, or edit their annotations after viewing the result of their annotations. The average time to annotate each frame is 90 seconds. The commands are represented with keyboard shortcuts and annotations are stored as clicks. This closely resembles the annotation workflow with which the Stathletes analysts are familiar.

The output is stored in JSON format, which is sent back to the University of Waterloo team once the annotations are complete. Upon receiving the output, the annotations are further verified by the researchers.

While Stathletes analysts have been using the annotation application, they have provided feedback and suggestions to the University of Waterloo developers. These suggestions have allowed for the annotations to be more accurate and the analysts to process more frames faster. For example, we added keyboard shortcuts to allow for the user to quickly delete pairs of annotations that add noise to the homography estimate. When the user reviews their annotation and chooses to redo the selected points, rather then having to start



(a) The warped frame overlayed on the rink model.

(b) The warped rink model overlayed on the frame.

Figure 3.3: During the verification stage of the homography annotation tool, the homography calculated from the point correspondences is visualized by warping the frame and rink model.

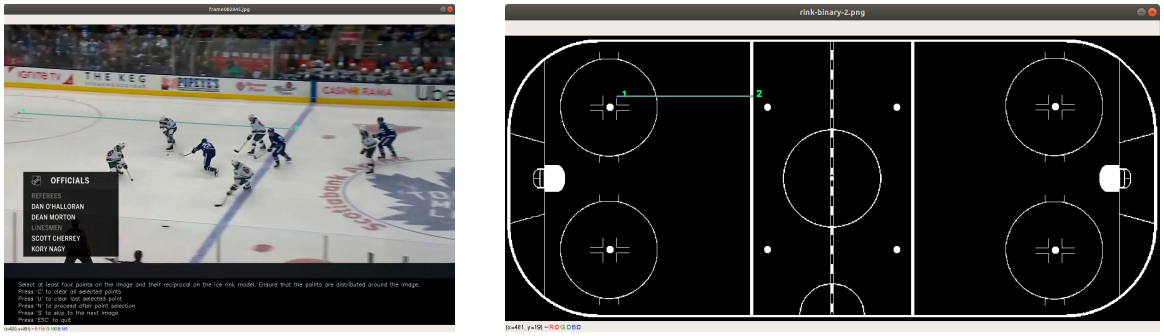
the annotation from the beginning, they are able to alter the points they have already selected.

While the broadcast frames were being annotated, there were certain frames that were the most difficult annotate well. These were the frames that were mostly filled by the area of the ice around either blue line. The lack of high fidelity keypoints made it difficult to obtain an accurate and robust annotation. A feature was added to allow the users to draw guidelines in order to obtain accurate annotations in areas of open ice (Fig. 3.4).

3.4 Description of Dataset

The hockey homography dataset collected with the annotation tool has 7,721 annotated frames from 24 separate game broadcasts from the 2018-19 NHL season. Two to three shots, or sequences of dynamic game play, were extracted from each game, each lasting approximately 30 seconds, for a total of 84 shots. All sequences were captured by the main broadcast camera, which is located in the stands above the centre line of the ice surface.

The videos were sampled at 1.5 frames per second (fps). The frames come from several different broadcasters, which means that the overlayed graphics are not the same across the dataset. A variety of home teams also means that there are different designs (e.g., team logo) embedded in the ice. Fig. 3.5 shows four sample frames from the dataset.



(a) Guideline drawn on the broadcast frame.

(b) Guideline drawn on the rink model.

Figure 3.4: Drawing guidelines on the broadcast frame and rink model. This allows the used to get a high fidelity annotation in an area with few keypoints. In this pair of frame and rink model, a guideline (turquoise) is drawn from the faceoff circle line (1) straight out to the blue line. A keypoint (2) can now be precisely annotated on the blue line where there would otherwise be no line intersections or endings.

In hockey broadcasts, the camera pans, tilts, and zooms to attempt to fill each frame with the play. This means that a significant portion of the ice surface is not shown in each frame [44]. However, when the whole dataset is aggregated, we obtain 100% coverage of the whole ice surface. Individually, the mean coverage of a frame is 24.1% of the ice surface. A histogram of the ice surface coverage for all frames is shown in Fig. 3.6.

The dataset is divided into train, test, and validation splits for research on ice rink localization methods randomly according to the games and venues from which the frames come. This ensures that the methods that memorize a rink appearance will obtain poor validation and test scores. Table 3.1 shows statistics about the splits of the dataset. Further details are shown in Appendix A.

Our dataset is to be used for training and testing methods for ice rink localization of NHL games. In the NHL there is a fixed set of venues, where every game is played in the home rink of one of the 31 teams. There are some exceptions to this rule, such as the NHL Winter Classic, where a game is played in an outdoor venue. However, we can assume that our methods only need to be used for use in the indoor rinks. An ideal dataset would therefore have games played in each of the 31 possible rinks. We could train and evaluate our methods on this dataset, since we would never be faced with an unknown rink. Another approach would be to train separate models with the same architecture for each of the rinks. In this way, there would be one model for each rink that works perfectly,

Figure 3.5: Four sample frames from the hockey homography dataset representing the diversity in rink appearances between games.

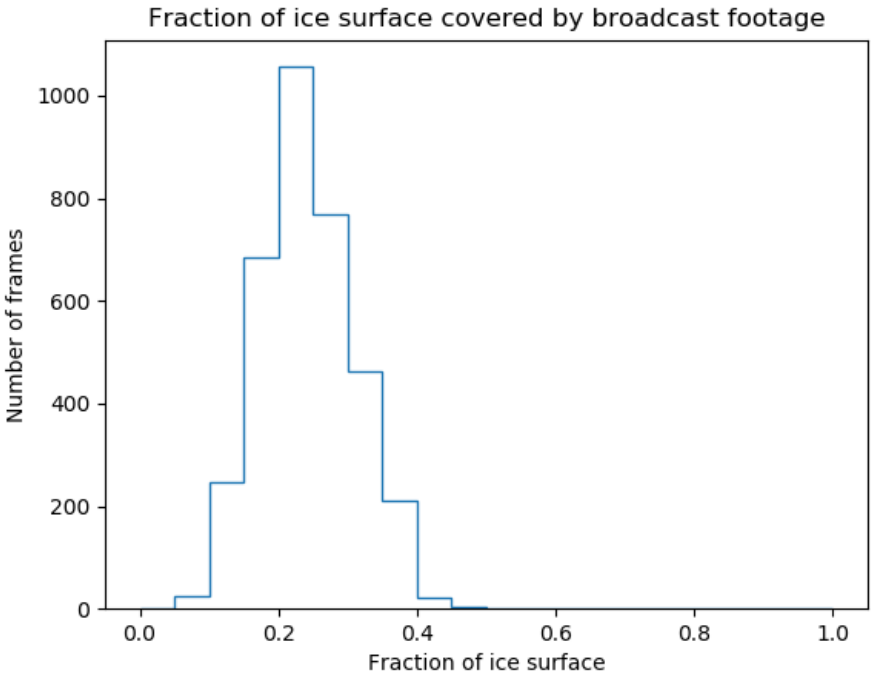


Figure 3.6: Distribution of ice surface coverage for all frames in the hockey homography dataset.

Split	Number of Games	Number of Frames
Train	21	6,524
Validation	2	706
Test	1	491
Total	24	7,721

Table 3.1: Statistics of the data splits from the hockey homography dataset.

rather than having to have a method that can generalize to all rinks in the league. Due to the difficulty of collecting annotations, our dataset contains games that were played in 13 rinks across the league.

During the annotation process, 450 frames were rejected due to the inability to get a satisfactory annotation. These frames are not included in the dataset descriptions in Tables 3.1 or A.1. Some failure modes for these frames included blur due to camera motion, occlusion by broadcast graphics, and too few high fidelity keypoints (Fig. 3.7).

3.5 Conclusion

We have developed a homography annotation tool to collect the transforms between frames from hockey broadcast video to an overhead rink model. The tool determines the ground truth transform through at least four point correspondences of landmarks on the ice. We also have collected a dataset of 7,721 hockey broadcast homographies. The dataset fills the gap of no publicly available ice rink localization datasets for research into broadcast camera calibration methods.

Figure 3.7: Failure modes for the hockey homography annotation tool. Clockwise from top left: ice blocked by broadcast graphic; too zoomed in, not enough high fidelity keypoints; motion blur; area around blue line, too few high fidelity keypoints.

Chapter 4

Highly Efficient Line Segmentation Deep Neural Network Architectures for Ice Rink Localization

Hockey analytics generated in real-time can be used by coaches and players to adapt their play to their opponents. It also allows for live data to be generated for sports betting and increased immersion for the fans in the game.

Inference from ice rink localization methods that rely on deep neural networks can be accelerated by making the network smaller. This means reducing the number of operations during inference. This chapter discusses two methods that use deep neural networks with a reduced number of parameters to segment the lines from the ice surface as an intermediate step for ice rink localization.

4.1 Introduction

Sports field localization is required to determine the absolute positions of the players and the puck on the ice, regardless of the broadcast camera's position. There have been methods developed to perform this analysis [17, 40, 45, 69], and several of these methods require segmentation of the lines on the playing field as an intermediate step. The resulting edge maps are used for further processing, such as for vanishing point estimation [40] or dictionary lookup [17, 69].

Despite the relative successes of these methods, there seems to be little focus into the selection of the semantic segmentation methods and justification for their use, but more into the downstream analysis [17, 40, 69]. This work deals solely with the line segmentation problem from hockey broadcast video, and proposes two efficient deep networks to solve it.

This work details two methods for approaching real-time performance for line segmentation from hockey broadcast video. BenderNet and RingerNet are small networks that achieve high accuracy on our annotated dataset from NHL games.

The contributions of this work are two lightweight semantic segmentation networks that effectively detect the lines on the playing surface from hockey broadcast video. BenderNet is two conditional generative adversarial networks (GANs) and RingerNet is segmentation network that uses dilated depthwise separable convolutions.

BenderNet achieves a mean intersection over union (mIOU) score of 31.12 with 2.8 million parameters and RingerNet achieves an mIOU score of 55.69 with 0.78 million parameters on the test split of the labelled dataset used in this work [52]. This opens the door for further research into small networks for line segmentation as an intermediate step for homography estimation.

4.2 Related Work

Works related to small semantic segmentation networks and sports field localization are reviewed here.

4.2.1 Line Segmentation for Sports Field Localization

In the literature, there have been several papers that attempt to solve the problem of sports field localization. Of these published methods, there are some that require segmenting the lines and outline of the ice surface as an intermediate step [17, 40, 69]. To our knowledge, there have not been any published methods that intensively explore the line segmentation component.

The sports field localization methods in the literature have some shortcomings, however, such as methods that only report performance on frames from soccer broadcast video [17, 69] and a lack of availability of the source code [69].

Line segmentation from broadcast soccer games differs from the same task with hockey for several reasons. First, the players are much smaller in relation to the field markings in

(a) Sample hockey broadcast frame.

(b) Sample soccer broadcast frame [40].

Figure 4.1: Sample hockey and soccer broadcast frames. Significant differences between the appearances in the two broadcasts means that sports field localization techniques cannot be directly transferred from one sport to the other.

soccer games than they are for hockey. The soccer playing field is much larger than the ice surface in a hockey rink and the broadcast camera tends to be further from the soccer field. This means that the broadcast camera for soccer games captures a larger area of the field. The typical position of the broadcast camera in professional hockey rinks means that the boards on the near side of the rink tend to be captured, which occludes the players and rink markings that are on the near side of the ice. The game of hockey is more dynamic and faster than that of soccer, which means that the broadcast camera for hockey pans and zooms faster and more often.

Finally, soccer games tend to be played outside, which means that they face variable lighting conditions depending on the configuration of the stadium and the time of day that the game is played. This may lead to some parts of the field in the shade and some players throwing a shadow. The variability can be within a stadium, depending on the time of day, and between stadiums with different configuration. Hockey games have a similar problem. Rinks across the NHL have different lighting configurations depending on their layout. Within one rink, there is also variability over the course of a game, despite the game being played indoors. Lights in the arena are hung from the rafters, directly over the ice surface, which can lead to relatively bright and dark areas on the ice. This can also be caused by shadows cast by the players skating on the ice. Furthermore, in-arena entertainment and photography can lead to flashes of bright light on the ice surface.

A comparison between hockey and soccer broadcast frames is shown in Fig. 4.1 For these reasons, methods for line segmentation that work well for soccer may not translate to an effective solution for hockey games.

Furthermore, in order to reduce inference time to approach real-time performance, the

Figure 4.2: Line segment detection is a different task than line segmentation in hockey, as it attempts to find edges of planar surfaces [9].

network to perform line segmentation should have few parameters [59]. VGGNet16 used by Homayounfar *et al.* [40] has 135 million parameters [70].

A method is needed that can achieve real-time or near real-time performance and also works well with a small training dataset.

4.2.2 Semantic Segmentation

Semantic segmentation is a dense prediction task in which each pixel in an input image is assigned to a class [19, 52]. Long *et al.* in 2015 were the first to demonstrate that fully convolutional networks, inspired by CNNs for other visual tasks, could be trained end-to-end with supervised pretraining and be used to obtain state-of-the-art results [52].

Recent semantic segmentation methods that achieve state-of-the-art segmentation accuracy use spatial pyramid pooling [19, 90]. These CNNs that achieve the highest accuracy typically require billions of FLOPs, hundreds of layers and thousands of channels [88]. There are several techniques that can make these large CNNs much smaller while maintaining acceptable accuracy, which include network compression, low-bit representation, and lightweight CNNs [57].

4.2.3 Line Segment Detection

The computer vision task of line segment detection has been approached as a unique area of research. This involves finding horizontal and vertical edges in a given scene that meet

at the camera's vanishing point [23]. The output of this task can be used for several downstream computer vision tasks, such as segmentation, depth estimation and 3D reconstruction [85]. The state of the art line segment detection method LETR uses a transformer architecture [85]. It holistically finds the line segments, rather than breaking the detection task into heuristic subtasks, as is normally done in line segment detection. LETR obtains a heatmap average precision (AP^H) of 86.3 on the Wireframe Dataset and 62.7 on the YorkUrban Dataset.

Line segmentation detection differs from the line segmentation task described in this paper, as they attempt find different lines in a given scene. Line segment detection looks for edges on the objects in the scene and these lines are straight and have zero width (i.e., can be defined as the connection between two points on the image) [23]. An example scene with annotated line segments is shown in Fig 4.2.Line segmentation for hockey rinks looks specifically for the lines on the hockey ice surface. These lines may be straight (e.g., blue line, goal line), curved (e.g., faceoff circle, goalie crease), or a circle (faceoff dots). The geometry of the lines on the hockey surface are known beforehand, due to the standard rink configuration in NHL games.

4.3 Methodology

The methodology for training and evaluating the line segmentation models is described in this section.

4.3.1 Dataset

The hockey homography dataset described in Chapter 3 was used for this method. Ground truth segmentations are extracted based on the ground truth homography [40]. The rink template was warped according to the ground truth homography and used as the segmentation mask. Three sample frames from the dataset and their associated line segmentation masks are shown in Fig. 4.3.

4.3.2 BenderNet

The lines are segmented from the ice in a two step process [17]. First, the playing surface is segmented from the boards and spectators, then the lines on the playing surface are

Figure 4.3: Three frames from the hockey line segmentation dataset. Occlusions from the near side boards can be seen in the left and centre frames and from overlayed broadcast graphics can be seen in the right frame. Pixels belonging to the line class are in blue.

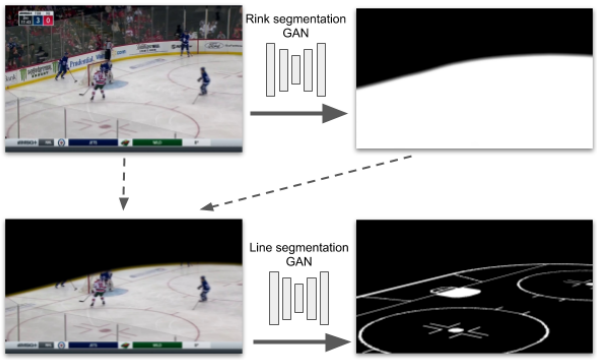


Figure 4.4: BenderNet architecture. The output of the rink segmentation GAN is used as a mask on the original frame. This combined image is used as input to a line segmentation GAN that isolates the lines.

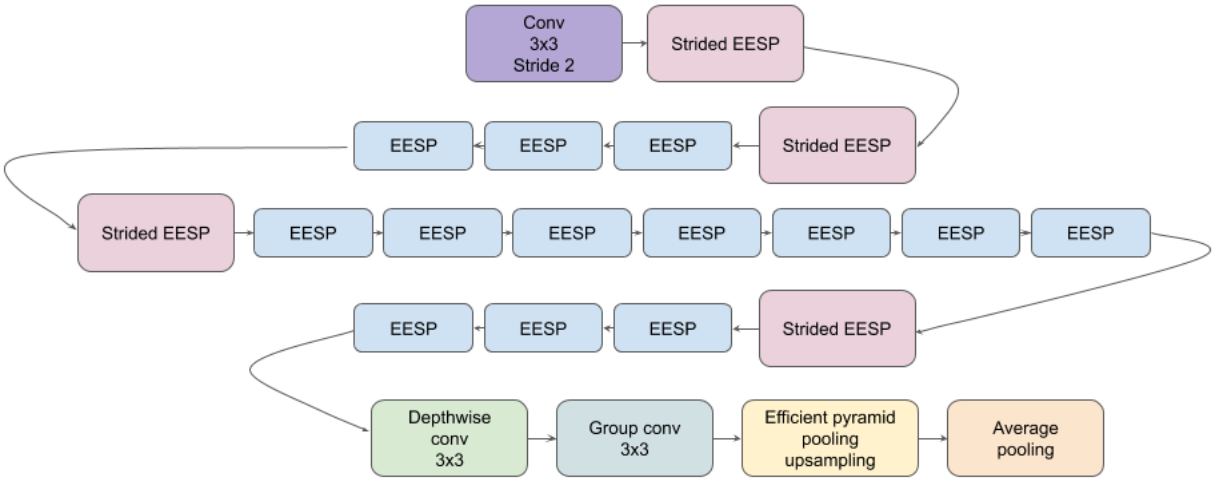


Figure 4.5: Network architecture of RingerNet. The EESP and Strided EESP modules extremely efficient spatial pyramid of depthwise dilated separable convolutions.

segmented from the masked frame. Both steps use simultaneously trained conditional adversarial networks [41]. Performing segmentation in two steps prevents any confusion from line-like structures on the boards or in the crowd [17].

BenderNet's architecture is based on Isola *et al.*'s pix2pix conditional adversarial network, for use in image translation tasks [41]. Line segmentation in this context can also be thought of as an image translation task, where the model of the rink is warped so that it overlays the lines in the frame. For both GANs, the generator and discriminators are U-Net shaped. The architecture of BenderNet is shown in Fig. 4.4.

4.3.3 RingerNet

A network to achieve real-time performance for line segmentation from hockey broadcast video was developed by extending an ESPNetv2 segmentation backbone [57]. The use of dilated depthwise convolutional blocks allows for a reduction in the size of the network without a large reduction in accuracy.

In a standard convolution, the number of parameters that must be learned is $n^2 c\hat{c}$, where c is the number of input channels, $n \times n$ is the size of the effective receptive field, and $\hat{c}$ is the number of output channels.

In depth-wise dilated separable convolutions, the convolution operation is factored into

two steps: 1) depth-wise dilated convolutions and 2) point-wise convolution. The first convolutions are performed on each input channel with a dilation rate of r, which gives a receptive field of $n_r \times n_r$, where $n_r = (n-1)(r+1)$. In the second convolution step, a linear combination of the channels is learned. This reduces the number of parameters that must be learned to $n^2c + c\hat{c}$.

RingerNet has an architecture that comprises alternating strided EESP (extremely efficient spatial pyramid of depthwise dilated separable convolutions) and EESP modules, as described by [57]. The architecture of the segmentation network is shown in Fig. 4.5.

4.4 Experimental Setup

Semantic segmentation of lines in hockey broadcast video with RingerNet and BenderNet were evaluated on the annotated hockey dataset. The results are reported in Table 4.1. Train and validation splits were obtained as described in Chapter 3. This ensures that there are games present in both splits that occur in different arenas, thereby allowing for validation of the method in varying lighting conditions and with different broadcasters.

4.4.1 BenderNet

The input to each GAN is two 256×256 images. For the rink segmentation GAN, the inputs are the original frame and a mask of the playing surface. For the line detection GAN, the inputs are the frame with everything other than the playing surface masked out and the line segmentation mask.

The two GANs have a U-Net backbone [67] and are trained for 100 epochs with a binary cross entropy loss. Training is also done with Adam optimizer with an initial learning rate of 0.0002 and momentum of 0.5. This is similar to the training parameters used by Chen and Litte [17].

Experiments: Performance of BenderNet is evaluated on two tasks: 1) line segmentation with the 2GAN architecture, and 2) rink segmentation with the first segmentation network.

The line segmentation test evaluates the performance of both GANs sequentially to first mask out the rink surface, then the lines on the playing surface. The performance is measured as the mIOU of the lines compared to the ground truth line segmentation mask.

We then determine the performance of the rink segmentation network individually (i.e., the first of the two GANs). The network is evaluated by calculating the mIOU with the ground truth segmentation masks of the rink surface.

The two GANs are trained sequentially, with the rink segmentation GAN trained on full broadcast video frames from the test split with ground truth segmentation masks for the ice surface. The second line segmentation GAN is trained on broadcast frames where all pixels other than the ice surface have been masked out. The ground truth data is segmentation masks for the lines.

4.4.2 RingerNet

RingerNet performs line segmentation in one step. It is based on ESPNetv2 and its use of dilated depthwise separable convolutions.

Training is performed with cross entropy loss and stochastic gradient descent with momentum and weight decay as the optimizer. Momentum is set to 0.9 and weight decay is 4×10^{-5}. A hybrid learning rate scheduler, as described by Mehta *et al.* [57], varies the learning rate during training. Initial learning rate is 0.009 and has 61 epochs of a cyclic learning rate policy before switching to linear.

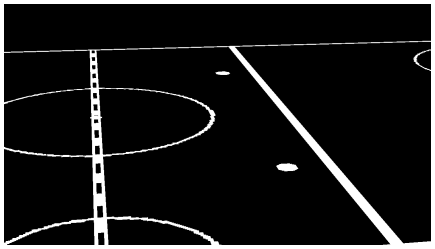
The network has an additional scale parameter, which is a scaling factor for the number of channels used throughout the model. In this segmentation model, the scale parameter was 2. All lines are assigned to the same class and the rink and stands are assigned to the background. Both classes are weighted evenly.

Training is performed in two steps. First the network is trained on 256×256 images, then 384×384 images [57].

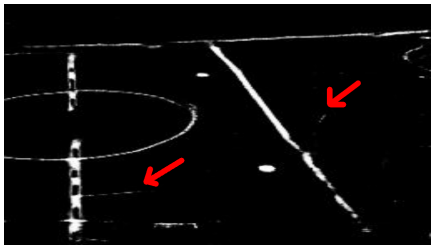
Experiments: The performance of RingerNet is evaluated on two tasks: 1) line segmentation on the frames directly extracted from the broadcast video, and 2) line segmentation on frames with the ice surface pre-segmented.

In the second experiment, the effects of preprocessing the video frames on the performance of the segmentation network were observed. This was inspired by BenderNet, in which the playing surface is segmented before performing detection on the lines. This line segmentation is performed with the ground truth annotations of the rink surface, where the spectators and boards are masked out before the frame is fed into the network.

(a) Input frame



(b) Ground truth line segmentations



(c) Line segmentation with BenderNet

(d) Line segmentation with RingerNet

Figure 4.6: Typical results for line segmentation with RingerNet and BenderNet. Inference is performed on a single frame from a broadcast video of an NHL hockey game. The red arrows in the BenderNet output show spurious detections.

4.5 Results and Discussion

Results for all experiments are reported in Table 4.1. The Task column refers to the experiment being performed. The rink segmentation task for BenderNet uses the first of the two GANs, which segments the ice surface from the broadcast frame. The rink and line segmentation task uses both GANs, first to segment the ice surface from a broadcast frame, then to segment the lines on the ice. The rink and line segmentation task for RingerNet segments the lines from a broadcast frame in one step. The line segmentation task for RingerNet takes in broadcast frames where everything other than the ice surface is masked out according to the ground truth homography and attempts to segment the lines.

BenderNet performs segmentation of the lines in two stages. In the first step, the playing surface is segmented from the surrounding area and achieves an mIOU of 98.96. Performance is significantly reduced in the second step, where the lines are segmented from the masked frame and an mIOU of 31.12 is obtained. This shows that ice segmentation is an easier problem than line segmentation, likely due to occlusions and shadows from the players and the fact that the lines occupy a small area on the ice surface.

RingerNet trained and tested on the broadcast video frames was able to obtain an

Table 4.1: Network sizes and performances of three segmentation methods on the NHL broadcast video dataset. The results in **bold** are the best results in network size and segmentation performance for the rink seg. + line seg. task.

Method	Task	Parameters [M]	mIOU
Homayounfar *et al.* [40]	line seg.	135 [70]	-
BenderNet	rink seg. + line seg.	2.8	31.12
BenderNet	rink seg.	2.8	98.96
RingerNet	rink seg. + line seg.	**0.78**	**55.69**
RingerNet	line seg.	0.78	60.08

mIOU of 55.69, which is higher than the segmentation achieved with BenderNet. These are promising, as a much smaller network can be used to obtain acceptable results for further processing to estimate homography. Sample output of RingerNet can be seen in Fig. 4.6d.

When analyzing the RingerNet results, a prior segmentation of the ice surface (i.e., masking out of the spectators and boards with ground truth rink mask) increases the performance of the line segmentation to an mIOU of 60.08. This shows that further accuracy gains can be obtained by preprocessing the frames to have a prior knowledge of the rink surface. An interesting next step would be to combine the highly accurate rink segmentation component of BenderNet and the efficient line segmentation method of RingerNet.

The reported mIOU of RingerNet is higher than that of BenderNet. The frames inferred with BenderNet tend to have more continuous lines, but it is easily confused by players and other markings on the ice. BenderNet has more discontinuous lines, but can discriminate better from distracting elements of the frames. Since the frames come from broadcast video, there may be additional graphics included, such as the game clock, advertisements, and scores from other games. RingerNet does a better job of avoiding these regions, even without having to initially segment the playing surface, as with BenderNet.

In Fig. 4.6c, the red arrows show spurious detections in the BenderNet output. While the lines are more continuous in this figure, as compared to the RingerNet output in Fig. 4.6d, these spurious detections may be more deleterious in the downstream processing methods [17, 40, 69]. For example, feature extraction, as performed in [17, 69] can include these regions. These spurious regions likely lead to the lower mIOU score for this method. To achieve better performance with BenderNet, a further post-processing step may be required to remove these methods before extracting features.

These results are encouraging to perform further experiments to investigate small networks for line segmentation. Further research could be done to observe the effects considering temporal aspects, since the input is a video.

4.6 Conclusion

BenderNet and RingerNet are two lightweight deep segmentation networks that achieve state-of-the-art results on the rink and line segmentation task for hockey broadcast video. In addition, a review of methods for performing sports field localization was performed, and there are several methods that require line segmentation as an intermediate step. The fundamental differences between hockey and other broadcast sports, especially soccer, means that a different, task-specific approach needs to be taken. The two efficient methods proposed in this paper can be used in sports field localization pipelines to achieve low-latency inference.

Chapter 5

Extracting Camera Parameters from Homography Transform Matrix

In hockey broadcast footage, the camera parameters of the broadcast camera are not made available. These intrinsic and extrinsic parameters must therefore be estimated from the inferred homography of each frame. The homography matrix represents the transform between each hockey broadcast frame and an overhead view. This chapter details a method for extracting camera parameters from the inferred homography of each frame in a broadcast video sequence with the objective of smoothing the homography estimates.

5.1 Motivation

Our baseline solution for inferring homography from hockey broadcast video is a model based on the ResNet-18 network architecture [37]. In our implementation, the fully connected layer has been replaced to output eight values, which are used to parameterize the homography transform. Details about the performance of this model are reported in Chapter 6.

Despite the high performance in accuracy of this method, when the output is visualized (i.e, by warping broadcast frames onto the rink model by the inferred homography), there are perturbations. This occurs due to the network inferring the homography for each frame individually, rather than making sure that the whole camera path is smooth.

Preliminary testing has shown that directly smoothing elements in the homography matrix does not give good results. Each frame's homography transform is not unique [34] and is overparameterized, compared to the actual motion of the broadcast camera.

5.2 Background

This section describes methods in the literature for decomposing the homography matrix into camera parameters for sports field localization.

5.2.1 Proposed Methods

There are two works in the literature that describe methods for extracting camera parameters from the homography matrix in a sports field localization application [68, 21].

Sha *et al.* propose a method to regress the homography of each of the frames from a broadcast video of a basketball game. It uses a neural network with three modules: semantic segmentation, camera pose initialization, and homography prediction with a spatial transform network [68]. In the camera pose initialization method, a dictionary of camera poses from the range of possible values in the training dataset is generated. For each frame in the training dataset, the focal length, 3D camera position, and pan and tilt angles are estimated with the method described by Zhang [89].

This method is interesting because it leverages many known priors to accurately estimate the homography of the image. Taking into account the possible camera poses could prove advantageous for estimation. The results reported in the paper show that the inference time is very fast, at 0.004 seconds. The reported accuracy is less than methods compared in the paper. The authors explain that this is due to a lack of training data.

Citraro *et al.* directly try to solve for the camera parameters from broadcast video of sports games, and uses the homography matrix as an intermediate step [21]. This method is based on keypoint segmentation: using a U-Net [67] to segment the intersections of lines on the playing surface as seen in each frame of the video.

This method seems to work quite well at a fast inference time. The authors report an inference time of 8 fps. The accuracy is also quite high, although this method did not perform well when applied to our hockey dataset. The methods for decomposing the homography matrices into camera parameters are very similar between the two methods, whereby the focal length is extracted before the pan and tilt angles. Neither method had an official implementation released for decomposing the homography matrix.

5.2.2 Camera Parameters in Sports Broadcast Video

Cameras in sports broadcast video can be represented as a pinhole camera with three degrees of freedom: focal length, pan angle, and tilt angle [17].

The projection of the points in the 3D world can be mapped to the 2D image world using the projection matrix P, which is 3×4. The projection matrix contains information about the extrinsic and intrinsic camera parameters [34]. Chen and Little show that the projection matrix can be shown as in Eq. 5.1.

$$P = KQ_\phi Q_\theta S[I| - C] \tag{5.1}$$

$KQ_\phi Q_\theta$ represents the pan, tilt, and zoom of the camera, where K is the intrinsic matrix of the camera, Q_ϕ is the camera's pan rotation, and Q_θ is the camera's tilt rotation. $S[I|-C]$ is a prior (i.e., constant for the camera in the sequence of frames), and S represents the rotation from world to the camera base and C is the translation to the camera's centre.

5.2.3 Smoothing Homography Based on Camera Parameters

Sharma *et al.* propose a method for stabilizing their homography estimation by optimizing an energy function parameterized by the pan angle, zoom angle, the centre of the camera, and the top and bottom intercepts of the quadrilateral of the frame projected onto the rink model [69]. By parameterizing the optimization problem in this way, the resulting smoothed camera motion function is similar to the way in which the broadcast camera operator captures the game [69].

5.3 Camera Parameter Extraction

This section details several methods that attempt to decompose the homography matrix into the camera parameters. There are no ground truth camera parameters against which to compare these results.

All of the methods in this section use the homographies that have been regressed with a modified ResNet-18 network that has been trained on our hockey homography dataset.

5.3.1 Focal Length

The first camera parameter to extract is the focal length, which is the zoom of the camera. Because the camera is far away from the ice surface, the focal length is quite large to capture what is happening in the game.

Focal Length from Homography Parameters

The first method for determining the focal length involves solving for the focal length directly from the elements of the homography matrix [21, 89]. The focal length can either be inferred using Eq. 5.2 or 5.3.

$$f^2 = -\frac{h_2 h_5}{h_0 h_3 + h_1 h_4} \tag{5.2}$$

$$f^2 = \frac{h_5^2 - h_2^2}{h_0^2 + h_1^2 h_3^2 - h_4^2} \tag{5.3}$$

The focal lengths calculated using this method for each frame in a sequence are shown in Fig. 5.1. The homographies from which this plot originates were regressed using the network with modified ResNet-18 architecture described earlier in this chapter. The estimates of the focal lengths were then smoothed with a median filter. This plot shows several gaps, which occurred where $f^2 < 0$.

The failure of the focal length calculation to reliable estimate real-valued focal lengths means that another method is required to parameterize the zoom of the camera.

Distance to Focal Point

To obtain a more reliable estimate of the focal length, the distance to the focal point was estimated for each frame. The homography matrix provides a view of how the broadcast camera sees the rink. Lines that are parallel in the frame converge when the frame is warped to the overhead view, as seen in Fig. 5.2. In the view from the broadcast camera, the left and right sides of the frame are parallel. However, when warped to the overhead view, the image takes on the shape of a trapezoid. If the left and right sides of the frame are extended, they converge at the focal point. The convergence of the two sides of the trapezoid at the focal point is shown in Fig. 5.3.

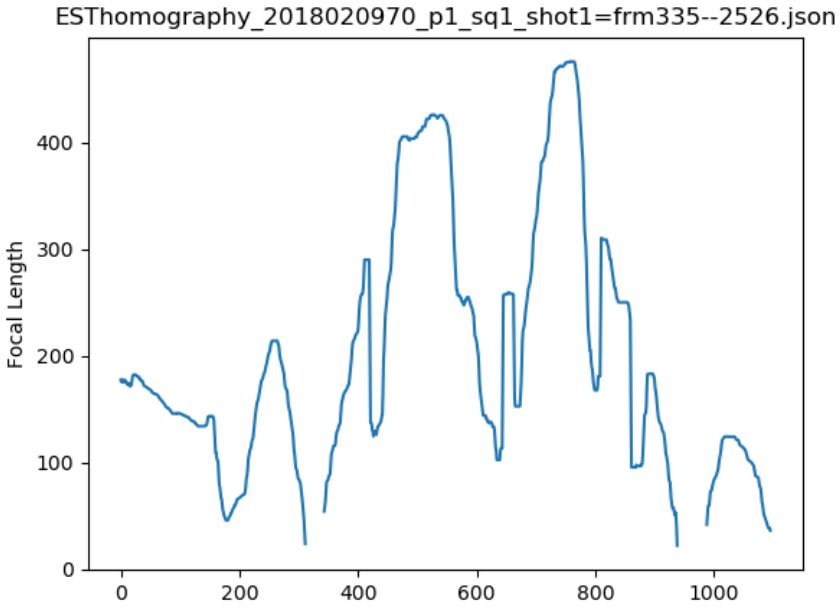


Figure 5.1: Focal length calculated with the elements of the homography matrix for the game sequence `2018020970_p1_sq1_shot1=frm335--2526` (Eq. 5.2). A median filter with $n = 21$ is also applied to smooth the data.

Figure 5.2: Hockey broadcast frame warped onto the rink model. This provides an overhead view of the broadcast frame according to the homography transform.

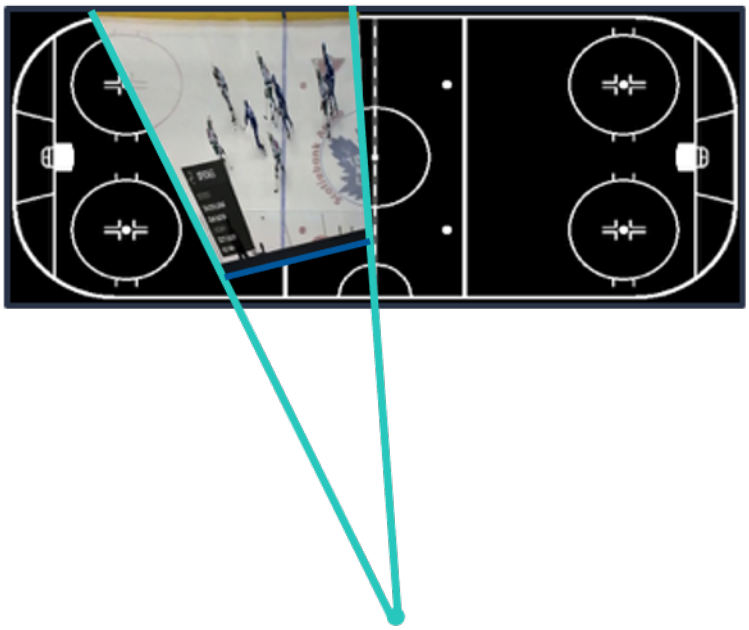


Figure 5.3: Overhead view of a hockey broadcast frame with the focal point. The focal point is determined as the point where parallel lines in the broadcast frame converge in the broadcast view.

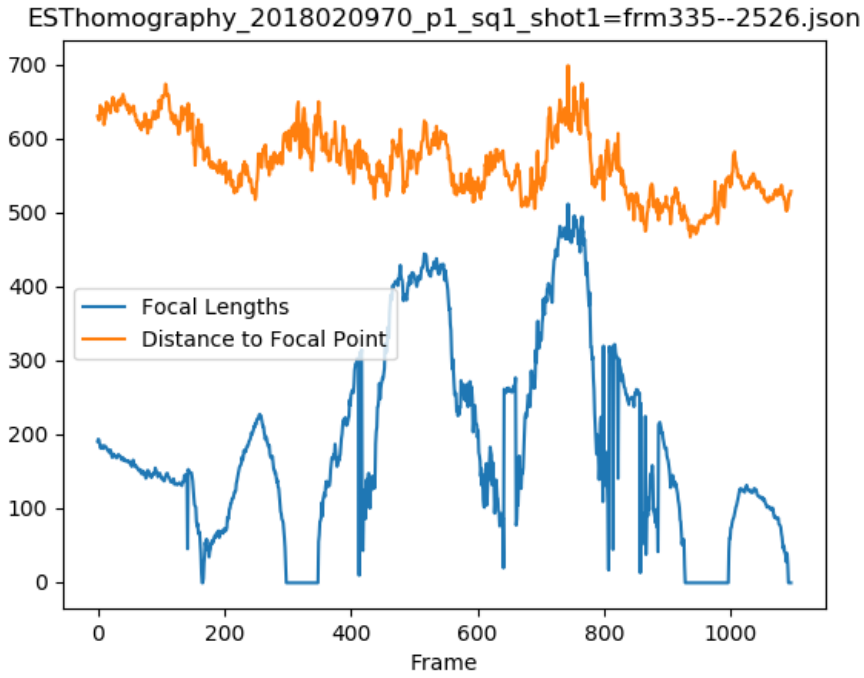


Figure 5.4: Distance to focal point and focal length (Eq. 5.2) calculated with the elements of the homography matrix for the game sequence `2018020970_p1_sq1_shot1=frm335--2526`.

The distance to the focal point is calculated as the length from the bottom of the frame (dark blue line in Fig. 5.3) to the focal point (turquoise dot at the bottom of Fig. 5.3). In the pinhole camera model, the distance to the focal point is proportional to the focal length of the model.

Fig. 5.4 shows a plot of the distance to the focal point (orange) and focal length calculated with Eq. 5.2 or 5.3 (blue). The two plot lines follow a very similar trajectory, showing that focal length are correlated. The distance to focal point measure is more reliable, as it does not have any discontinuities in the plot.

Normalized Focal Length

Despite the distance to focal length metric seeming to be a good substitute for the focal length, an even better estimate of the focal length can also be determined with the geometry of the frame warped onto the rink model. A normalized focal length can be determined

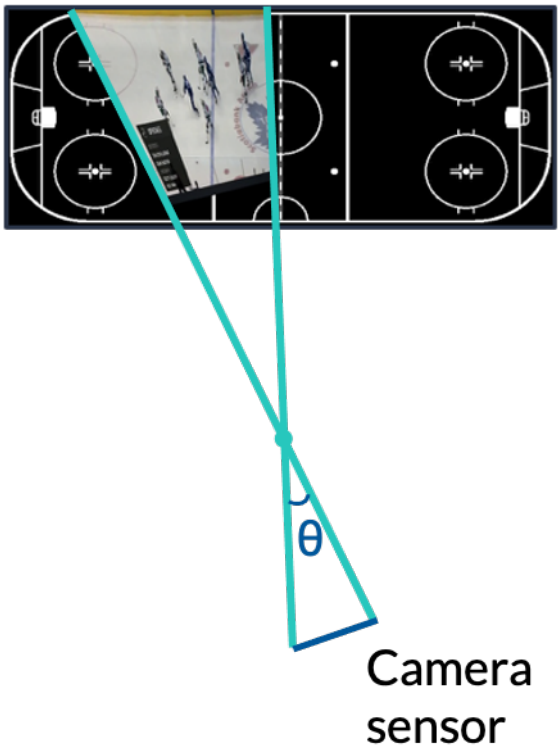


Figure 5.5: Geometry for determining the normalized focal length from the overhead view of a broadcast frame. The normalized focal length is the distance between the camera sensor and the focal point (turquoise dot).

using the geometry as in Fig. 5.5. The focal length is the distance between the camera sensor (dark blue line) and the focal point (turquoise dot).

Using trigonometry, $\tan(\theta/2) = (w/2)/f$, where θ is the angle that subtends the convergence of the two sides of the broadcast frame projected onto the rink model, f is the focal length, and w is the width of the camera sensor. We can solve for the focal length as in Eq. 5.4.

$$f = \left(\frac{w}{2}\right) \frac{1}{\tan\left(\theta/2\right)} \tag{5.4}$$

Over the course of the video sequence, the focal length changes, but the camera sensor will maintain the same width. Therefore, we can calculate a normalized focal length f' as in Eq. 5.5.

$$f' = \frac{2}{w} f = \frac{1}{\tan(\theta/2)} \tag{5.5}$$

The angle θ can be calculated with the cosine rule and the length of the sides of the triangle formed by the two sides of the warped broadcast frame and the bottom of the frame. The normalized focal length and distance to the focal point for one sequence of frames are shown in Fig. 5.6.

Comparing the normalized focal length to the distance to focal point shows that their plots take on a very similar shape. This shows that these two measures are related. Normalized focal length gives the correct focal length of the broadcast camera up to a scale that is particular to the broadcast camera's sensor width.

5.3.2 Pan Angle

As hockey is a dynamic game, the play transitions frequently between the two teams' ends and the neutral zone. The camera pans to follow this motion across the ice.

The pan angle of the broadcast camera is defined as the angle between a line on the overhead view of the broadcast frame that extends from the centre of the top edge to the centre of the bottom edge, and a vertical line that extends through the middle of the rink model, as seen in Fig. 5.7. Fig. 5.8 show a plot of pan angle for a game sequence.

5.3.3 Tilt Angle

Extracting the tilt angle from the homography matrix proved to be much more difficult than the focal length and pan angle. An initial estimate was parameterized as the angle of the centre line of the camera from the horizontal. This angle is shown as θ in Fig. 5.9. The angle is defined such that when the camera's centre line is pointed at the opposite end of the rink, the tilt angle is 30 degrees. This assumption was made based on observations of the position of the broadcast camera in the rink and approximate geometry of the rink. A precise pan angle measurement would require knowing the vertical and horizontal positions of the camera above the rink surface.

The plot for tilt angle in degrees for a game sequence is shown in Fig. 5.10. The range of the tilt angle is quite small, and is centered around 46 degrees. This behaviour is expected, as the angle being greater than 30 degrees means that middle of the rink is within the field of view. The tilt angle increases toward the end of the sequence to follow play that is at the near side boards.

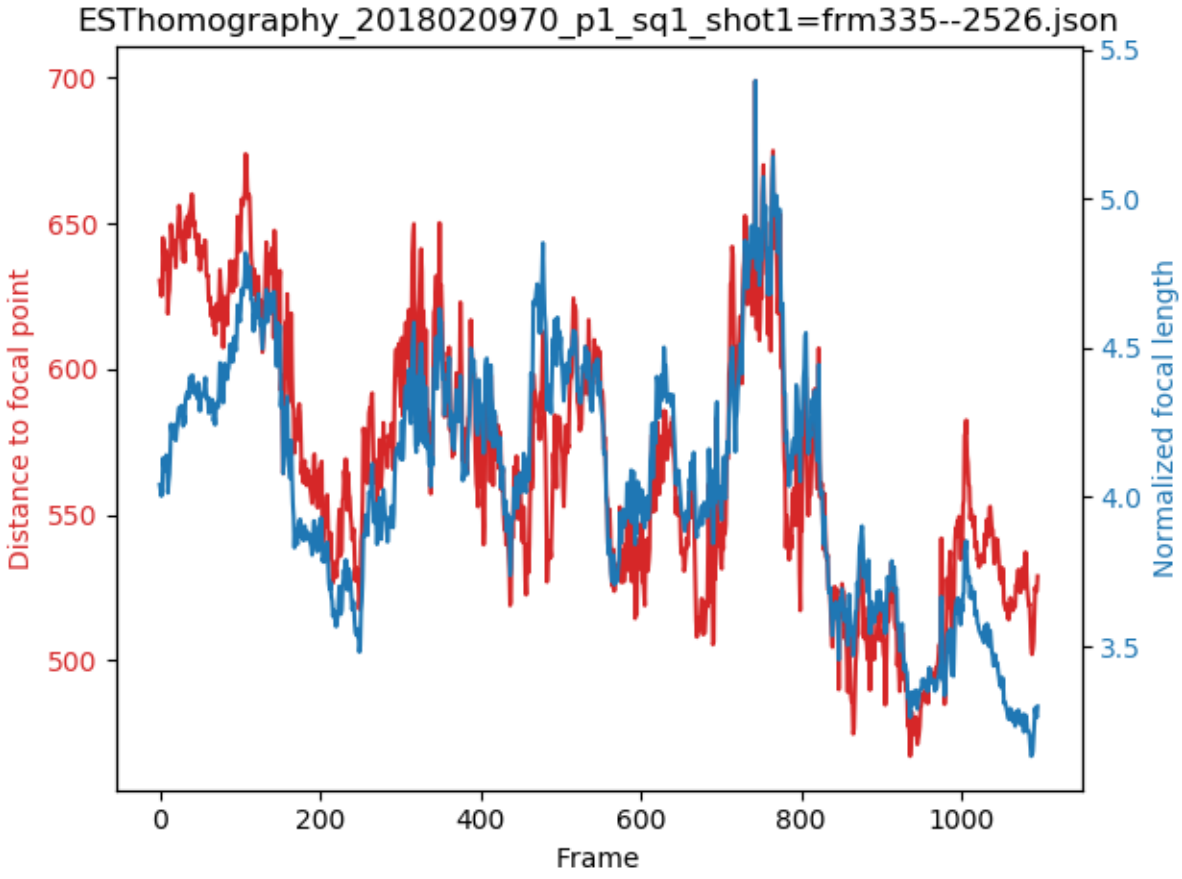


Figure 5.6: Normalized focal length and distance to focal point for the game sequence `2018020970_p1_sq1_shot1=frm335--2526`.

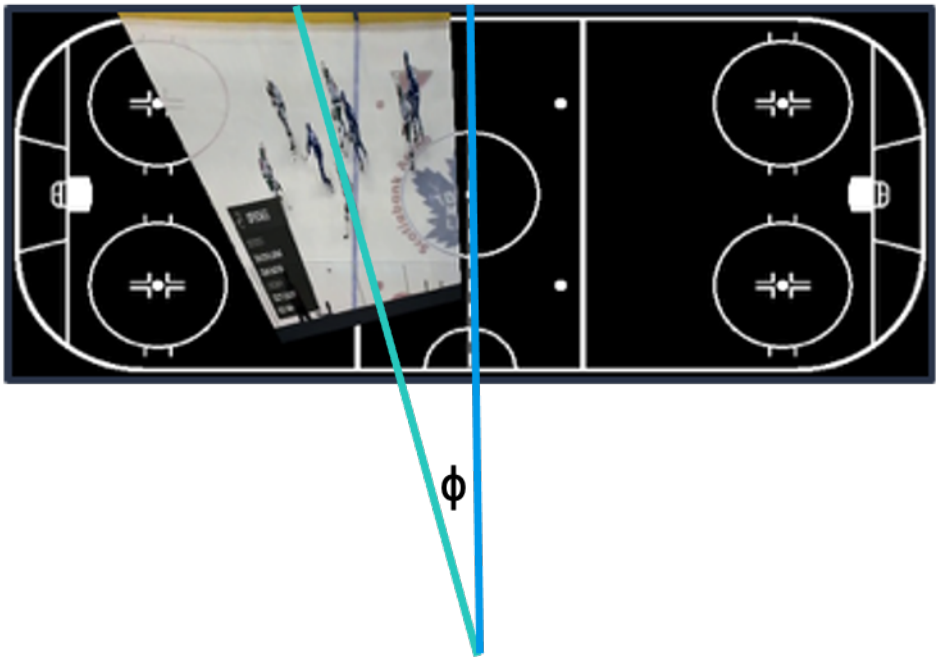


Figure 5.7: Geometry for determining the pan angle ϕ of the overhead view of the broadcast frame.

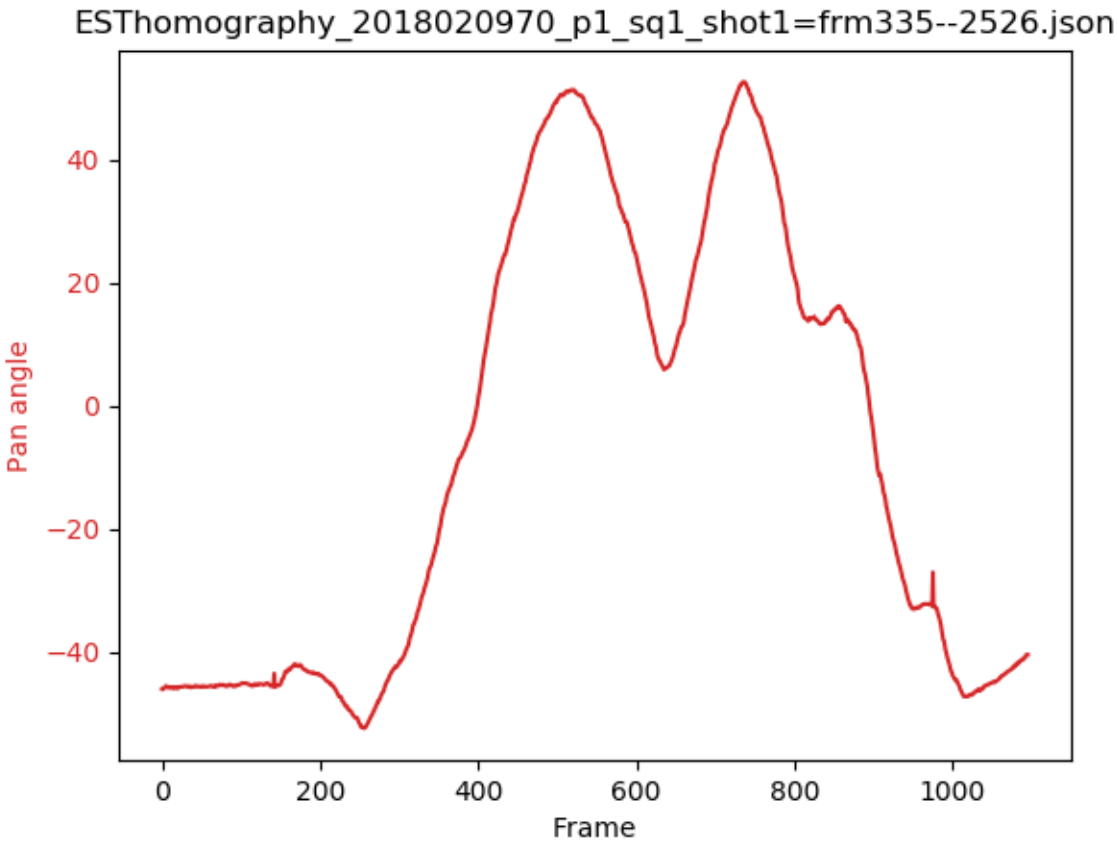


Figure 5.8: Pan angle in degrees calculated for the game sequence `2018020970_p1_sq1_shot1=frm335--2526`.

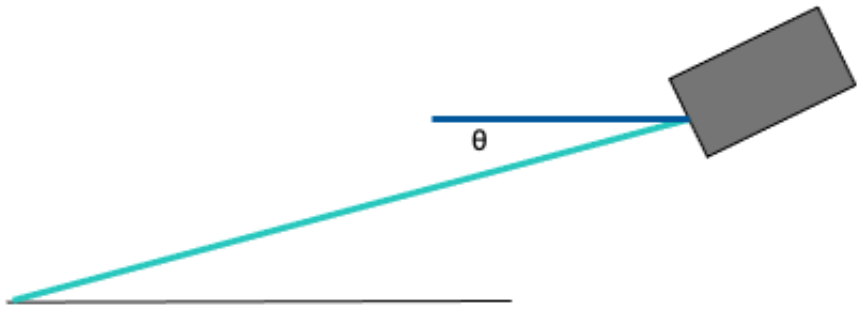


Figure 5.9: Geometry for determining the tilt angle θ of the overhead view of the broadcast frame. When the centre line of the camera is pointing directly across the rink, the tilt is assumed to be 30 degrees.

5.4 Conclusion

The homography matrix that defines the warp of each frame in a hockey broadcast video to the overhead rink model can be represented with three free camera parameters: pan, tilt, and zoom [17]. State-of-the-art sports field localization methods regress the homography transform for each frame in the sequence. To ensure that the output of the sports field localization method is temporally continuous, a smoothing method is required. The camera parameters that have been extracted using the methods described in this chapter can be smoothed and the homography matrix reconstructed, as a next step. This will give an effective way to localize the sports field.

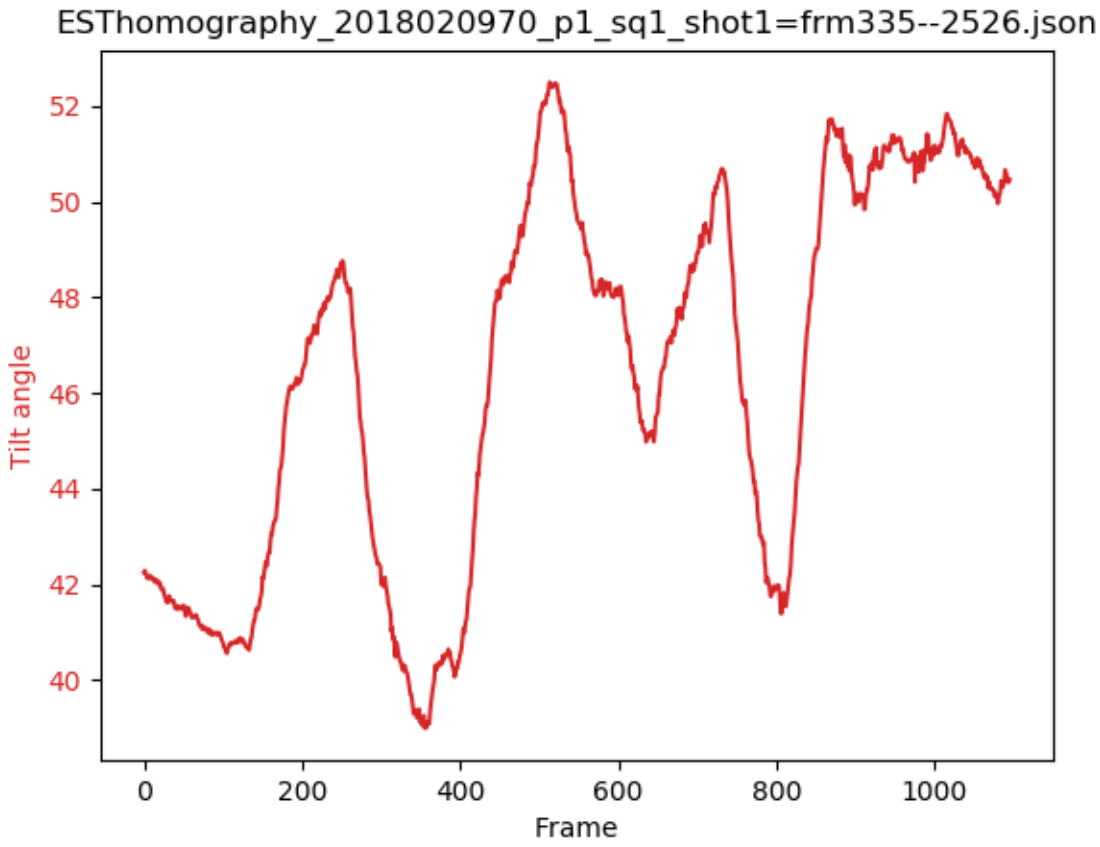


Figure 5.10: Tilt angle in degrees calculated for the game sequence `2018020970_p1_sq1_shot1=frm335--2526`.

Chapter 6

Simultaneous Sports Field Localization and Smoothing

Existing sports field localization pipelines tend to infer the parameters of the broadcast camera in a frame-by-frame basis [17, 21, 63, 69]. If the inference method were to work well, when hockey broadcast video frames are warped onto the overhead rink model according to the inferred homography matrices, the resulting video should be smooth, similar to the smoothness of the input video. Due to errors in the inference method, there may be some perturbations in the output, whereby the output video does not appear smooth. Many techniques add a smoothing step after the initial sports field localization estimation to ensure the best parameters are inferred to represent the motion of the broadcast camera [17, 21, 63, 69]. This approach is explored in Chapter 5.

Rather than inferring the homography for each frame in a sequence individually, then smoothing the output afterwards, this chapter attempts to develop a method that uses a deep network architecture to infer and smooth the broadcast camera parameters in one step.

6.1 Background

This section reviews some methods in the literature that use deep networks to refine sports field localization estimates and techniques for video analytics.

6.1.1 Deep Networks for Sports Field Localization Refinement

Many datasets in the literature are very sparse and may not include temporal information about the frames that have been selected. This makes training and testing methods that rely on neighbouring frames for sports field localization not feasible. This section discusses deep neural networks that have been used to refine sports field localization estimates.

In their method for sports field localization, Sha *et al.* use the localization layers of a spatial transform network (STN) to refine an initial homography estimate retrieved from a database of training samples [68]. The localization network takes in the input frame and the selected frame from the dictionary, both of which have had their zones segmented, and returns the parameters of the transform between the two frames [42]. The transform is parameterized with the 8 parameters from the homography transform [68].

Jiang *et al.* propose a method for regressing the homography transform by optimization [45]. After inferring an initial estimate of the broadcast camera parameters, a separate neural network is used to evaluate the registration error of the warped frame on the model. The registration error is backpropagated to the homography parameters to obtain the gradient, which is used to update the gradient. This process is performed iteratively, until a threshold is reached.

These two methods use deep networks to improve an initial homography estimate, however they do take advantage of the redundancy that is found across frames in a video sequence. Furthermore, the sports field localization datasets that are described in the literature use point correspondences to collect the ground truth homography matrices for each of the frames (see Chapter 3). Despite the care that the analysts who annotate these sequences use, there is still some error present in the annotated frames, since the true camera parameters are not available. The methods for sports field localization in the literature do not attempt to tackle this problem.

6.1.2 Video Analytics

The tasks of object detection and semantic segmentation on single images are quintessential computer vision tasks, however, extending these techniques to videos presents additional challenges. Rather than acquiring frame-level detections or segmentations and then a post-processing step to join the frames into a video, there are several methods in the literature that attempt to perform object tracking and semantic segmentation in a single step.

Feichtenhofer *et al.* propose a detect and track approach to take in a sequence of video frames and perform multi-task learning for frame-based object detection and across-frame

track regression based on a region-based fully convolutional network [28]. Bergmann *et al.* transform an object detection network into a tracktor framework by using the regression head for bounding box refinement to perform temporal realignment of bounding boxes [12]. Meinhardt *et al.* propose a TrackFormer architecture, that uses transformers to attend to the objects in the scene [58].

Oh *et al.* tackle the problem of video object segmentation, where the segmentation map is given for the first frame in a sequence and segmentation maps must be estimated for all other frames [64]. They propose a neural network that computes the spatio-temporal attention on each pixel for the whole sequence.

These approaches show that exploiting the spatio-temporal information from the videos can lead to improved results, as compared to only frame-wise approaches.

6.2 Experiments

This section discusses several network architectures for simultaneous sports field localization and smoothing.

6.2.1 Heatmap-Type Architecture

The uncertainty in the hockey broadcast video dataset due to annotation via point correspondences likely leads to a bias when attempting to evaluate methods. Rather than directly attempting to regress homography parameters, a heatmap approach is used [79].

To regress the localization of a given broadcast frame, the homography transform is parameterized as the coordinates of points on the broadcast frame warped onto the overhead rink model. Fig. 6.1 shows the location of the control points on the broadcast video frame (orange) and the locations of the control points after they have been warped according to the ground truth homography transform H (blue). The homography regression network regresses the positions of the blue dots. The orange points are normalized, such that they are the same for each frame that passes through the pipeline [45]. They are located at the bottom left and right corners of the frame, and on the left and right sides at a height of $0.6 \times$ the height of the frame.

If the height and width of the frame were normalized to 1, respectively, the coordinates of the control points would be $(0, 1)$ (bottom left), $(1, 1)$ (bottom right), $(0, 0.6)$ (top left) and $(1, 0.6)$ (top right).

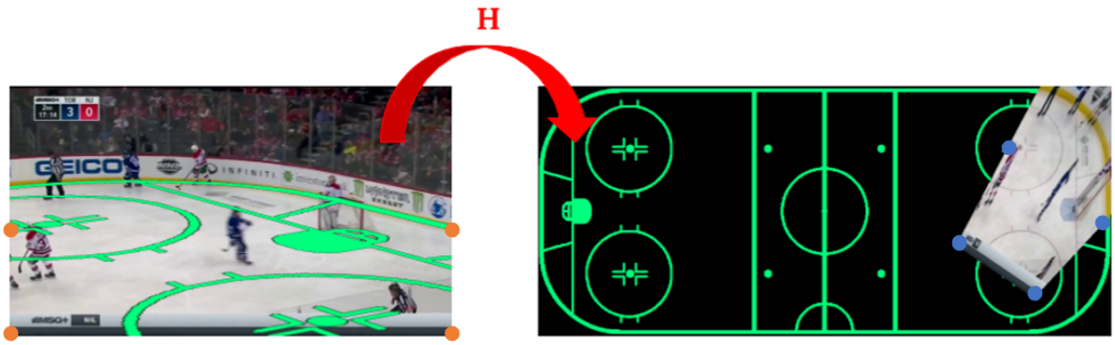


Figure 6.1: Locations of the control points on the broadcast frame (orange) and warped onto the rink model (orange) according to H. The broadcast frame localization network regresses the coordinates of the blue points.

To account for uncertainty in the ground truth annotations, instead of regressing the coordinates of the control points, a network is used to estimate a heatmap for each of the control points. The network outputs four heatmaps, for each of the control points. Each of the heatmaps predicts the probability of the control point occuring at each pixel.

Network Design

The network is based on the ResNet-18 architecture for image classification [37]. The fully connected layer was removed and the feature map was upsampled to the output size. Then convolutional layers were added to reduce the number of channels to 4. The output size is varied to observe the effects on the performance of the network. The network architecture is shown in Fig. 6.2.

Training Configuration

The ground truth data for this architecture is generated by determining the locations of the control points on the rink model. A Gaussian distribution is then drawn on the rink model with standard deviation of 5% of the width of the rink model and mean that is the location of the control point. The sizes of the heatmaps were varied to observe the effects on the performance.

Training was performed on our hockey rink localization dataset for 100 epochs with mean squared error loss and Adam optimizer with a learning rate of 0.0001.

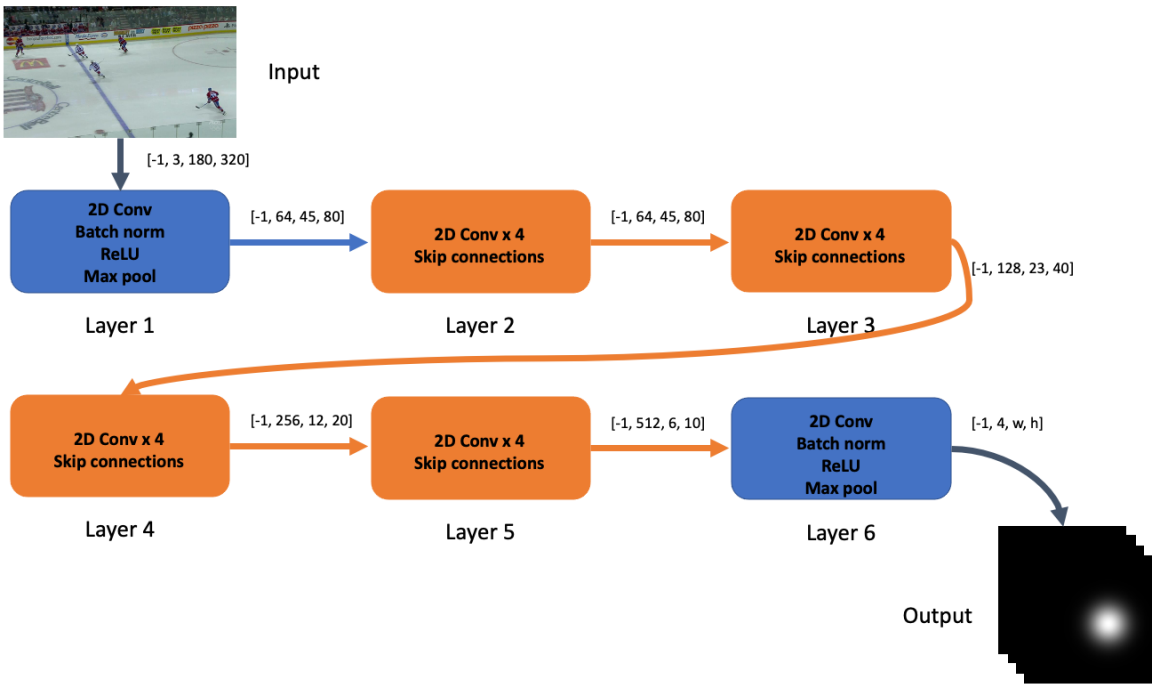


Figure 6.2: Architecture of the heatmap-type architecture based on the ResNet-18 architecture.

Table 6.1: Performance of the heatmap-type architecture for hockey rink localization. Directly regressing the positions of the warped control points has a heatmap of size none. The accuracy score is the percentage of correctly localized keypoints.

Heatmap Size	Accuracy	IOU_{part}	$\text{IOU}_{\text{whole}}$	Mean Smoothness
None	61.72	89.81	9.20	463.50
64×64	11.14	57.30	2.20	467.96
128×128	15.37	57.15	2.42	510.10
256×256	13.74	55.01	1.07	486.74

Results

The performance of the heatmap-type architecture and directly regressing the coordinates of the control points is reported in Table 6.1. The mean smoothness score is the average of the smoothness of the four control points. Smoothness is calculated for a sequence of points by calculating the standard deviation of the differences of each of the coordinates. For this score, lower is better.

A test point is considered to be correctly localized at a tolerance of 5% of the dimensions of the rink if the L2 distance between the ground truth keypoint location x_{gt} and predicted keypoint location x_{pred} is less than $t = \begin{bmatrix}10 & 4.25\end{bmatrix}$ ft. This means that $\|x_{gt} - x_{pred}\|_2 < t$. The accuracy score is the percentage of correctly localized keypoints at a tolerance of 5% of the dimensions of the rink.

None of the results from the heatmap-type architectures outperformed directly regressing the coordinates of the control points. Interestingly, despite having a significantly lower IOU_{part} and IOU_{part}, the method using a 64×64 heatmap has a mean smoothness that is close to the mean smoothness of the network that directly regressed the coordinates of the control points. This result is interesting, that a similar heatmap-based network that obtains higher accuracy, potentially with different training parameters, could improve the smoothness of the control point coordinate regression.

6.2.2 Long Short-Term Memory Network

Long short-term memory networks (LSTMs) are a type of network architecture that can take in sequential data [39]. They are advantageous because they are able to learn long-term dependencies, which allows for input from a previous time step to be used in the computation for a current time step.

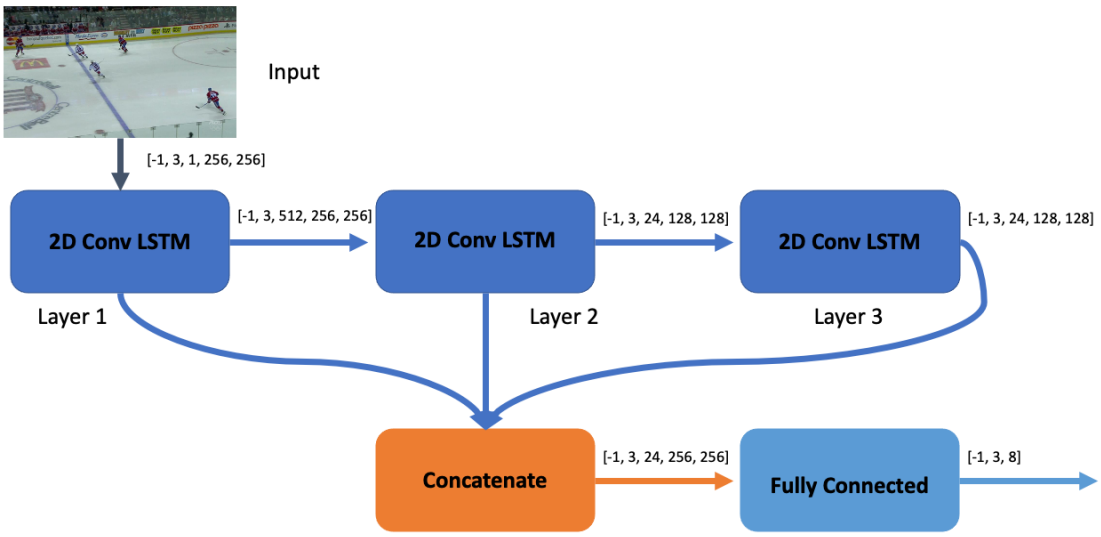


Figure 6.3: Architecture of the multi-scale LSTM architecture for hockey rink localization.

Using LSTMs for hockey rink localization would allow for the network to consider features from previous frames in the broadcast video sequence. Allowing the network to have access to richer input could allow for a more accurate homography inference.

Network Design

The network is multi-scale and uses 2D convolutional LSTM modules. The architecture is shown in Fig. 6.3.

Training Configuration

Sequences of three frames were used as input and the homographies, parameterized as the coordinates of the four control points, for all frames in the sequence were the output.

The model was trained for 100 epochs with mean squared error loss function and Adadelta optimizer with a learning rate of 0.001.

Table 6.2: Performance of the LSTM network architecture for hockey rink localization.

IOU_{part}	IOU_{whole}
41.24	13.14

Results

The performance of the LSTM architecture for hockey rink localization is reported in Table 6.2. The only metrics that were captured for this architecture are IOU_{part} and IOU_{whole}.

These IOU scores are higher than the other temporal models explored in this chapter. Evaluating the smoothness of these architectures would be beneficial to allow for an accurate comparison.

6.2.3 Temporal Convolutions

Temporal convolutional networks use convolutions in the time dimension [49]. These provide an advantage over recurrent networks and LSTMs because they are better able to capture long-range patterns. These networks are also faster to train and perform better than LSTMs. Using this type of architecture for hockey rink localization is advantageous because it can allow for more features to infer the homography transform.

Network Design

The network takes in a sequence of three frames, the first two of which have known homographies and the third for which the homography will be estimated. First, features are extracted from all of the frames with a ResNet-18 network pretrained on ImageNet [37]. Next, the two frames with known homographies are warped onto the rink model and their features are extracted with ResNet-18 network pretrained on ImageNet. Each frame's ResNet features and corresponding warped frame's ResNet features are concatenated and 1D convolution is performed. These features and the ResNet-18 features of the frame with unknown homography are concatenated and 1D convolution is performed along the temporal axis. Finally, a fully connected layer outputs the locations of the control points of the input frame.

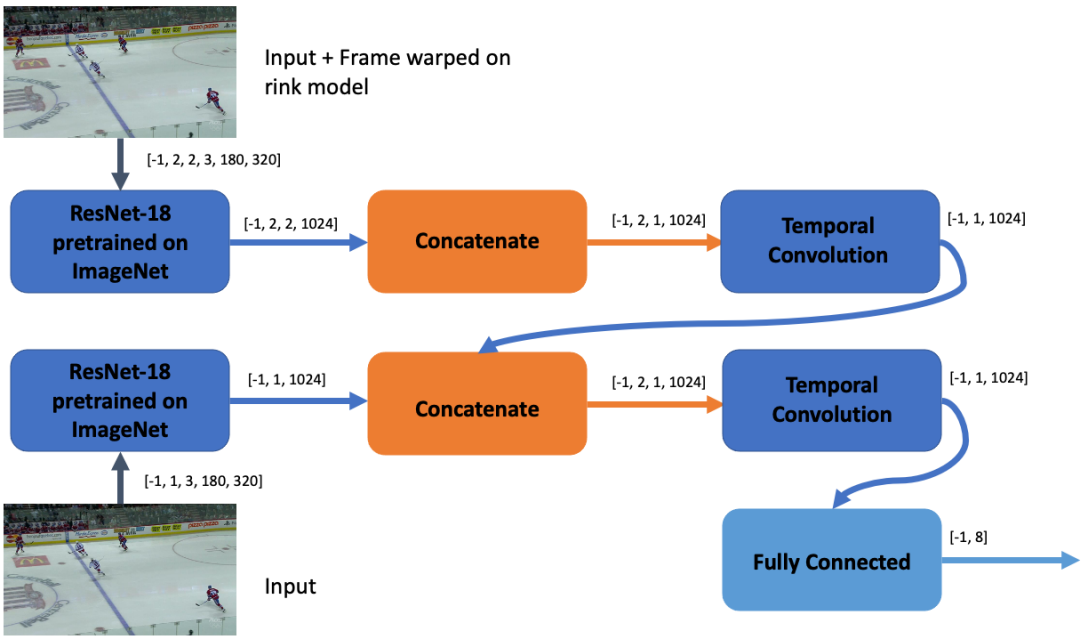


Figure 6.4: Architecture of the temporal convolutional architecture for hockey rink localization.

Table 6.3: Performance of the temporal convolutional network architecture for hockey rink localization.

Accuracy	IOU_{part}	IOU_{whole}	Mean Smoothness
0.79	16.1	0.03	356.07

Training Configuration

Sequences of three frames were used as input and the homographies, parameterized as the coordinates of the four control points, for the final frame in the sequence was the output.

The model was trained for 200 epochs with mean squared error loss function and Adam optimizer with an initial learning rate of 0.0001.

Results

The performance of the temporal convolutional network is reported in Table 6.3.

This network architecture is based on the ResNet-18 architecture because of its high performance for regressing the coordinates of the control points. This modified architecture has lower accuracy and IOU scores than the network that directly regresses the coordinates of the control points. This may be because this temporal convolutional architecture does not accurately model the temporal relationships between the frames. Higher accuracy could be obtained by increasing the receptive field of the temporal convolution layers and including more frames in the input sequence.

Despite the poor accuracy, the smoothness score is much lower. This means that the temporal convolutions increase the smoothness across a sequence of frames. This architecture could combine previously inferred control point coordinates to have a better prediction for the next frame in the sequence.

6.3 Conclusion

This chapter proposes three methods for simultaneously inferring the homography of frame from a hockey broadcast video while smoothing it with the surrounding frames. All three methods use deep neural networks to perform these tasks. Further research in this area can give a pipeline for sports field localization that does not require an additional postprocessing step to smooth the output.

Chapter 7

Conclusion

Ice rink localization is a fundamental step in the pipeline for automatic hockey analytics extraction from broadcast video. The broadcast camera operator pans, tilts, and zooms to fill the field of view of the camera with the play. To compensate for this motion, the camera must be calibrated. Despite the variety of methods in the literature for sports camera calibration, there is no one method that performs particularly well for all sports and applications.

This thesis presents a new hockey rink localization annotation tool and dataset. It also presents three research directions for improving ice rink localization methods. The lack of publicly available datasets contributes to the difficulty in developing solid models for sports field localization. This works presents a tool for collecting point correspondences to get the homography transform for frame in a hockey broadcast video to an overhead view of the rink. With this tool, a dataset of 7,721 frames was collected and used for the research on ice rink localization methods for the rest of the dissertation.

Using small neural networks to localize the rink can make inference faster. Methods for segmenting the lines on the playing surface can be used as a component in an efficient ice rink localization pipeline. The motion of the broadcast camera is smooth, and ice rink localization is usually performed on a frame-wise basis, which can result in unstable output. Smoothing the camera parameters over a sequence of frames more closely replicates the motion of the camera operator. The pan, tilt, and zoom can be extracted from each homography matrix using the geometry of the warped frame and smoothed. Finally, rather than regressing the ice rink localization and smoothing the camera parameters in two separate steps, deep neural network architectures can perform these two tasks simultaneously.

The results of this research provide a contribution to hockey analytics. The methods

presented in this paper can be used to improve the accuracy and decrease inference time for sports field localization, which is a fundamental element in an automatic hockey analytics pipeline.

7.1 Potential for Future Research

The research presented in this paper provides a basis for research into ice rink localization for broadcast hockey video. Some future research directions in this paper include increasing the size of the dataset and evaluating the methods within a full ice rink localization pipeline.

Future work with the homography annotation tool would be to expand the size of the dataset. While this dataset is relatively large, compared to other sports field localization datasets, a larger dataset would potentially allow for better solutions. Annotating hockey broadcast frames that have been sampled at a higher density would also allow for more research into methods that account for temporal information, particularly the deep networks that simultaneously regress ice rink localization and smooth.

The methods that have been presented in this paper work well on their own, however future research should focus on testing and developing these methods within an ice rink localization pipeline. For instance, the method for segmenting lines on the playing surface with a small network requires another step to localize the ice, such as a dictionary lookup [17, 68] or vanishing point estimation [40].

Once the camera parameters are extracted from each frame's homography matrix, a method for smoothing them and reconstructing the homography matrix needs to be developed. The neural networks that simultaneously regress the homography matrix and smooth presented in this dissertation should also be a starting point for further research to get improved performance for accuracy and smoothness. The method with temporal convolutions offers the most promise in this area.

7.2 Thesis Applicability

This thesis presents several approaches for localizing the rink in hockey broadcast video. Using the methods presented in the paper, a pipeline for homography estimation can be developed. This pipeline can be used to automatically extract hockey analytics from broadcast video. For example, with a method to detect the locations of players within a broadcast frame, the absolute locations of the players on the ice can be determined. This

can then be used to derive analytics about the players and the game, such as skating speed, and event localization.

7.3 Thesis Impact

The methods presented in this thesis in conjunction with a fully-realized ice rink localization pipeline can contribute to higher accuracy when automatically deriving analytics from broadcast hockey video. Reduced inference time means that analytics can be generated faster, potentially in real time and included in the broadcast. These analytics can also be used by players and staff of hockey teams to make data-driven decisions to improve the performance of their team. The use of widely available broadcast footage for automatically generating analytics means that more teams can use this technology, thereby increasing equity in hockey.

References

[1] Black Girl Hockey Club. https://blackgirlhockeyclub.org/. Accessed: 2021-01-14.

[2] Global $4.5 billion sports analytics market forecasts up to 2024. https://www.businesswire.com/news/home/20181205005823/en/Global-4.5-Billion-Sports-Analytics-Market-Forecasts. Accessed: 2019-09-10.

[3] Hockey Canada Annual Report, July 2019 - June 2020. Technical report. Accessed: 2021-01-14.

[4] Hockey Diversity Alliance. https://hockeydiversityalliance.org/. Accessed: 2021-01-14.

[5] NHL Currently Testing Sportlogiq as Optical Tracking Partner. https://www.sporttechie.com/nhl-testing-sportlogiq-optical-tracking-partner-data/. Accessed: 2019-09-10.

[6] Open Source Computer Vision Documentation. https://docs.opencv.org/master/index.html. Accessed: 2021-03-08.

[7] Professional Women's Hockey Player Association. https://pwhpa.com/. Accessed: 2021-01-14.

[8] PyInstaller. https://www.pyinstaller.org/. Accessed: 2021-03-08.

[9] E. J. Almazan, R. Tal, Y. Qian, and J. H. Elder. MCMLSD: A dynamic programming approach to line segment detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, July 2017.

[10] V. Machaca Arceda, K. Fernández Fabián, and J.C. Gutíerrez. Real time violence detection in video. *IET Conference Proceedings*, January 2016.

[11] H. Bay, T. Tuytelaars, and L. Van Gool. SURF: Speeded up robust features. In *Proceedings of the European Conference on Computer Vision*, pages 404–417. Springer, 2006.

[12] P. Bergmann, T. Meinhardt, and L. Leal-Taixe. Tracking without bells and whistles. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

[13] Y. Cai, N. de Freitas, and J. J. Little. Robust visual tracking for multiple targets. In *Proceedings of the European Conference on Computer Vision*, pages 107–118. Springer, 2006.

[14] Z. Cai, H. Neher, K. Vats, D. A. Clausi, and J. Zelek. Temporal hockey action recognition via pose and optical flows. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2019.

[15] P. Carr, Y. Sheikh, and I. Matthews. Point-less calibration: Camera parameters from gradient-based alignment to edge images. In *2012 IEEE Workshop on the Applications of Computer Vision (WACV)*, pages 377–384, January 2012. ISSN: 1550-5790.

[16] A. Chan, M. D. Levine, and M. Javan. Player identification in hockey broadcast videos. *Expert Systems with Applications*, 165:113891, Mar 2021.

[17] J. Chen and J. J. Little. Sports Camera Calibration via Synthetic Data. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 2497–2504, June 2019. ISSN: 2160-7516.

[18] J. Chen, F. Zhu, and J. J. Little. A two-point method for PTZ camera calibration in sports. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 287–295, 2018.

[19] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European Conference on Computer Vision*, pages 801–818, 2018.

[20] Y. Chen, Z. Wang, Y. Peng, Z. Zhang, G. Yu, and J. Sun. Cascaded pyramid network for multi-person pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7103–7112, 2018.

[21] L. Citraro, P. Márquez-Neila, S. Savarè, V. Jayaram, C. Dubout, F. Renaut, A. Hasfura, H. B. Shitrit, and P. Fua. Real-time camera pose estimation for sports fields. *Machine Vision and Applications*, 31(3):16, March 2020.

[22] C. Cuevas, D. Quilón, and N. García. Automatic soccer field of play registration. *Pattern Recognition*, 103:107278, July 2020.

[23] P. Denis, J. H. Elder, and F. J. Estrada. Efficient edge-based methods for estimating manhattan frames in urban imagery. In *Computer Vision – ECCV 2008*, pages 197–210, 2008.

[24] D. DeTone, T. Malisiewicz, and A. Rabinovich. Deep image homography estimation. *arXiv preprint arXiv:1606.03798*, 2016.

[25] X. Duan. Automatic determination of puck possession and location in broadcast hockey video. Master's thesis, University of British Columbia, Vancouver, BC, 2009.

[26] E. Dubrofsky and R. Woodham. Combining line and point correspondences for homography estimation. In *Proceedings of the 4th International Symposium on Advances in Visual Computing, Part II*, pages 202–213, December 2008.

[27] M. Fani, H. Neher, D. A. Clausi, A. Wong, and J. Zelek. Hockey action recognition via integrated stacked hourglass network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, July 2017.

[28] C. Feichtenhofer, A. Pinz, and A. Zisserman. Detect to track and track to detect. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3038–3046, 2017.

[29] D. A. Forsyth and J. Ponce. *Computer vision: a modern approach*. Prentice Hall Professional Technical Reference, 2002.

[30] W. Förstner, T. Dickscheid, and F. Schindler. Detecting interpretable and accurate scale-invariant keypoints. In *2009 IEEE 12th International Conference on Computer Vision*, pages 2256–2263, 2009.

[31] B. Ghanem, T. Zhang, and N. Ahuja. Robust video registration applied to field-sports video analysis. In *2012 IEEE International Conference on Acoustics, Speech, and Signal Processing*, January 2012.

[32] T. Guo, K. Tao, Q. Hu, and Y. Shen. Detection of ice hockey players and teams via a two-phase cascaded cnn model. *IEEE Access*, 8:195062–195073, 2020.

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

[34] R. Hartley and A. Zisserman. *Multiple view geometry in computer vision*. Cambridge University Press, 2003.

[35] J.-B. Hayet and J. Piater. On-line rectification of sport sequences with moving cameras. In *MICAI 2007: Advances in Artificial Intelligence*, volume 4827, pages 736–746, 2007.

[36] J.-B. Hayet, J. Piater, and J. Verly. Robust incremental rectification of sport video sequences. In *British Machine Vision Conference*, January 2004.

[37] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[38] R. Hess and A. Fern. Improved video registration using non-distinctive local image features. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, June 2007.

[39] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[40] N. Homayounfar, S. Fidler, and R. Urtasun. Sports field localization via deep structured models. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4012–4020, July 2017.

[41] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1125–1134, 2017.

[42] M. Jaderberg, K. Simonyan, A. Zisserman, and K. Kavukcuoglu. Spatial transformer networks. In *Proceedings of the 28th International Conference on Neural Information Processing Systems-Volume 2*, pages 2017–2025, 2015.

[43] A. Jain and D. K. Vishwakarma. State-of-the-arts violence detection using convnets. In *2020 International Conference on Communication and Signal Processing (ICCSP)*, pages 813–817, 2020.

[44] Michael Jamieson. Tracking players: What you see is what you get! https://www.youtube.com/watch?v=l6LGVRB2X8U, 2021. Simon Fraser University Sports Analytics Seminar.

[45] W. Jiang, J. C. G. Higuera, B. Angles, W. Sun, M. Javan, and K. M. Yi. Optimizing through learned errors for accurate sports field registration. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, March 2020.

[46] H. Kim and K.-S. Hong. Robust image mosaicing of soccer videos using self-calibration and line tracking. *Pattern Analysis and Applications*, 4:9–19, March 2001.

[47] A. Kuznetsov and A. V. Savchenko. A new sport teams logo dataset for detection tasks. In *Computer Vision and Graphics*, pages 87–97, 2020.

[48] H. Le, F. Liu, S. Zhang, and A. Agarwala. Deep homography estimation for dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7652–7661, 2020.

[49] C. Lea, M. D. Flynn, R. Vidal, A. Reiter, and G. D. Hager. Temporal convolutional networks for action segmentation and detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 156–165, 2017.

[50] W. Li, Z. Wang, B. Yin, Q. Peng, Y. Du, T. Xiao, G. Yu, H. Lu, Y. Wei, and J. Sun. Rethinking on multi-stage networks for human pose estimation. *arXiv preprint arXiv:1901.00148*, 2019.

[51] G. Liu, X. Tang, D. Sun, and J. Huang. Robust registration of long sport video sequence. In *International Conference on Computer Vision Systems: Proceedings (2007)*, 2007.

[52] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.

[53] D. G. Lowe. Object recognition from local scale-invariant features. In *Proceedings of the seventh IEEE international conference on computer vision*, volume 2, pages 1150–1157, 1999.

[54] K. Lu, J. Chen, J. J. Little, and H. He. Lightweight convolutional neural networks for player detection and classification. *Computer Vision and Image Understanding*, 172:77 – 87, 2018.

[55] W. Lu, K. Okuma, and J. J. Little. Tracking and recognizing actions of multiple hockey players using the boosted particle filter. *Image and Vision Computing*, 27(1):189 – 205, 2009. Canadian Robotic Vision 2005 and 2006.

[56] W. McNally, K. Vats, A. Wong, and J. McPhee. EvoPose2D: Pushing the boundaries of 2D human pose estimation using neuroevolution. *arXiv preprint arXiv:2011.08446*, 2020.

[57] S. Mehta, M. Rastegari, L. Shapiro, and H. Hajishirzi. ESPNetV2: A light-weight, power efficient, and general purpose convolutional neural network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9190–9200, 2019.

[58] T. Meinhardt, A. Kirillov, L. Leal-Taixe, and C. Feichtenhofer. Trackformer: Multi-object tracking with transformers. *arXiv preprint arXiv:2101.02702*.

[59] P. Molchanov, S. Tyree, T. Karras, T. Aila, and J. Kautz. Pruning convolutional neural networks for resource efficient inference. *arXiv preprint arXiv:1611.06440*, 2016.

[60] S. Mukherjee, R. Saini, P. Kumar, P. P. Roy, D. P. Dogra, and B. Kim. Fight detection in hockey videos using deep network. *Journal of Multimedia Information System*, 4(4):225–232, 2017.

[61] H. Neher, K. Vats, A. Wong, and D. A. Clausi. Hyperstacknet: A hyper stacked hourglass deep convolutional neural network architecture for joint player and stick pose estimation in hockey. In *2018 15th Conference on Computer and Robot Vision (CRV)*, pages 313–320, 2018.

[62] T. Nguyen, S. W. Chen, S. S. Shivakumar, C. J. Taylor, and V. Kumar. Unsupervised deep homography: A fast and robust homography estimation model. *IEEE Robotics and Automation Letters*, 3(3):2346–2353, 2018.

[63] X. Nie, S. Chen, and R. Hamid. A robust and efficient framework for sports-field registration. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 1936–1944, January 2021.

[64] S. W. Oh, J.-Y. Lee, N. Xu, and S. J. Kim. Video object segmentation using space-time memory networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9226–9235, 2019.

[65] K. Okuma, J. J. Little, and D. Lowe. Automatic acquisition of motion trajectories: tracking hockey players. In Simone Santini and Raimondo Schettini, editors, *Internet Imaging V*, volume 5304, pages 202 – 213. International Society for Optics and Photonics, SPIE, 2003.

[66] H. Pidaparthy and J. Elder. Keep your eye on the puck: Automatic hockey videography. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1636–1644, 2019.

[67] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 234–241. Springer, 2015.

[68] L. Sha, J. Hobbs, P. Felsen, X. Wei, P. Lucey, and S. Ganguly. End-to-end camera calibration for broadcast videos. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13624–13633, June 2020.

[69] R. A. Sharma, B. Bhat, V. Gandhi, and C. V. Jawahar. Automated top view registration of broadcast football videos. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 305–313, March 2018.

[70] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[71] P. Skinner and S. Zollmann. Localisation for augmented reality at sport events. In *2019 International Conference on Image and Vision Computing New Zealand (IVCNZ)*, pages 1–6, December 2019.

[72] S. Tarashima. SFLNet: Direct sports field localization via CNN-based regression. In *Pattern Recognition*, pages 677–690. Springer International Publishing, 2020.

[73] G. Thomas. Real-time camera tracking using sports pitch markings. *Journal of Real-Time Image Processing*, 2:117–132, October 2007.

[74] G. Thomas, R. Gade, T. B. Moeslund, P. Carr, and A. Hilton. Computer vision for sports: Current applications and research topics. *Computer Vision and Image Understanding*, 159:3–18, 2017.

[75] S. Tian. Group event recognition in ice hockey. Master's thesis, University of British Columbia, Vancouver, BC, 2016.

[76] M. R. Tora, J. Chen, and J. J. Little. Classification of puck possession events in ice hockey. In *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 147–154, 2017.

[77] H. Tsurusaki, K. Nonaka, R. Watanabe, T. Konno, and S. Naito. Sports camera calibration using flexible intersection selection and refinement. *ITE Transactions on Media Technology and Applications*, 9(1):95–104, 2021.

[78] K. Vats, M. Fani, P. Walters, D. A. Clausi, and J. Zelek. Event detection in coarsely annotated sports videos via parallel multi-receptive field 1d convolutions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2020.

[79] K. Vats, W. McNally, C. Dulhanty, Z. Q. Lin, D. A. Clausi, and J. Zelek. Pucknet: Estimating hockey puck location from broadcast video. *arXiv preprint arXiv:1912.05107*, 2020.

[80] K. Vats, H. Neher, D. A. Clausi, and J. Zelek. Two-stream action recognition in ice hockey using player pose sequences and optical flows. In *2019 16th Conference on Computer and Robot Vision (CRV)*, pages 181–188, 2019.

[81] V. Viswanathan. Why AI is the next frontier in sports fan engagement and revenue. https://www.forbes.com/sites/forbestechcouncil/2019/08/16/why-ai-is-the-next-frontier-in-sports-fan-engagement-and-revenue. Accessed: 2019-09-10.

[82] W. Lu and J. J. Little. Simultaneous tracking and action recognition using the pca-hog descriptor. In *The 3rd Canadian Conference on Computer and Robot Vision (CRV'06)*, pages 6–6, 2006.

[83] P. B. Walters, D. Clausi, and A. Wong. Sports field localization using memory networks. *Journal of Computational Vision and Imaging Systems*, 5(1):2–2, 2019.

[84] P. Wen, W. Cheng, Y. Wang, H. Chu, N. C. Tang, and H. M. Liao. Court reconstruction for camera calibration in broadcast basketball videos. *IEEE Transactions on Visualization and Computer Graphics*, 22(5), May 2016.

[85] Y. Xu, W. Xu, D. Cheung, and Z. Tu. Line segment detection using transformers without edges. 2021.

[86] Q. Yao, A. Kubota, K. Kawakita, K. Nonaka, H. Sankoh, and S. Naito. Fast camera self-calibration for synthesizing free viewpoint soccer video. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1612–1616, 2017.

[87] R. Zeng, R. Lakemond, S. Denman, S. Sridharan, C. Fookes, and S. Morgan. Calibrating cameras in poor-conditioned pitch-based sports games. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1902–1906, April 2018.

[88] X. Zhang, X. Zhou, M. Lin, and J. Sun. Shufflenet: An extremely efficient convolutional neural network for mobile devices. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6848–6856, 2018.

[89] Z. Zhang. A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11):1330–1334, 2000.

[90] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia. Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2881–2890, 2017.

APPENDICES

Appendix A

List of Games in Hockey Homography Dataset

The hockey homography dataset contains 7,716 frames from 24 games during the 2018-19 NHL season. Table A.1 contains information about the games in the dataset and Table A.2 contains information about the games in the splits of the dataset.

Table A.1: Games in the hockey homography dataset.

Game ID	Period	Date	Home Team	Away Team	Number of Frames
2018020621	3	2019-01-03	Toronto Maple Leafs	Minnesota Wild	647
2018020639	2	2019-01-05	Toronto Maple Leafs	Vancouver Canucks	227
2018020652	1	2019-01-07	Toronto Maple Leafs	Nashvile Predators	120
2018020672	2	2019-01-10	Toronto Maple Leafs	New Jersey Devils	217
2018020690	2	2019-01-12	Toronto Maple Leafs	Boston Bruins	271
2018020706	3	2019-01-14	Toronto Maple Leafs	Colorado Avalanche	227
2018020729	3	2019-01-17	Tampa Bay Lightning	Toronto Maple Leafs	380

2018020733	3	2019-01-18	Florida Panthers	Toronto Maple Leafs	196
2018020754	3	2019-01-20	Toronto Maple Leafs	Arizona Coyotes	228
2018020765	1	2019-01-23	Toronto Maple Leafs	Washington Capitals	244
2018020904	2	2019-02-17	Pittsburgh Penguins	New York Rangers	159
2018020907	1	2019-02-17	Detroit Red Wings	Philadelphia Flyers	491
2018020916	3	2019-02-19	Florida Panthers	Buffalo Sabres	361
2018020941	2	2019-02-22	Detroit Red Wings	Minnesota Wild	351
2018020967	1	2019-02-25	Vancouver Canucks	Anaheim Ducks	255
2018020965	2	2019-02-25	Nashville Predators	Edmonton Oilers	283
2018020966	3	2019-02-25	Colorado Avalanche	Florida Panthers	391
2018020963	3	2019-02-25	New Jersey Devils	Montreal Canadiens	315
2018020970	1	2019-02-26	Philadelphia Flyers	Buffalo Sabres	450
2018021133	3	2019-03-20	Washington Capitals	Tampa Bay Lightning	340
2018021145	3	2019-03-21	Edmonton Oilers	Columbus Blue Jackets	233
2018021144	1	2019-03-21	Calgary Flames	Ottawa Senators	416
2018021163	2	2019-03-23	Los Angeles Kings	Anaheim Ducks	236
2018021150	3	2019-03-23	New Jersey Devils	Arizona Coyotes	678

Table A.2: Games in the hockey homography dataset splits.

Train Game IDs	Validation Game IDs	Test Game IDs
2018020621	2018020966	2018020907
2018020639	2018020963	
2018020652		
2018020672		
2018020690		
2018020706		
2018020729		
2018020733		
2018020754		
2018020765		
2018020904		
2018020916		
2018020941		
2018020965		
2018020967		
2018020970		
2018021133		
2018021144		
2018021145		
2018021150		
2018021163		