PuckNet: Estimating hockey puck location from broadcast video

Kanav Vats William McNally Chris Dulhanty Zhong Qiu Lin
David A. Clausi John Zelek
University of Waterloo
Waterloo, Ontario, Canada
{k2vats,wmcnally,chris.dulhanty,zhong.q.lin,dclausi,jzelek}@uwaterloo.ca

Abstract

Puck location in ice hockey is essential for hockey analysts for determining the location of play and analyzing game events. However, because of the difficulty involved in obtaining accurate annotations due to the extremely low visibility and commonly occurring occlusions of the puck, the problem is very challenging. The problem becomes even more challenging in broadcast videos with changing camera angles. We introduce a novel methodology for determining puck location from approximate puck location annotations in broadcast video. Our method uniquely leverages the existing puck location information that is publicly available in existing hockey event data and uses the corresponding one-second broadcast video clips as input to the network. The rationale behind using video as input instead of static images is that with video, the temporal information can be utilized to handle puck occlusions. The network outputs a heatmap representing the probability of the puck location using a 3D CNN based architecture. The network is able to regress the puck location from broadcast hockey video clips with varying camera angles. Experimental results demonstrate the capability of the method, achieving 47.07% AUC on the test dataset. The network is also able to estimate the puck location in defensive/offensive zones with an accuracy of greater than 80%.

Introduction

Ice hockey is played by an estimated 1.8 million people worldwide (IIHF 2018). As a team sport, the positioning of the players and puck on the ice are critical to offensive and defensive strategy (Thomas 2006). Currently, practical methods for tracking the position of each player and the puck for the full duration of a hockey match are limited. Advances in computer vision have shown promise in this regard (Lu, Okuma, and Little 2009; Pidaparthy and Elder 2019), but ultimately remain in the developmental phase. As an alternative, radio-frequency identification is currently being explored for player and puck tracking (Cavallaro 1997; Gulitti 2019), but may only be financially and logistically feasible at the most elite professional level, e.g., the National Hockey League (NHL). Information regarding player and puck position is therefore inaccessible in most cases. As a result, the conventional heuristic approach for evaluating the effectiveness of team strategies involves analyzing the record of *events* that occurred during the match (turnover, shot, hit, face-off, dump, etc.) (Tora, Chen, and Little 2017; Fani et al. 2017).

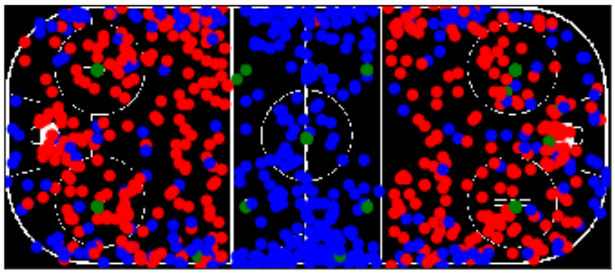

Figure 1: Distribution of some puck locations on the hockey rink. The locations are evenly distributed throughout the ice rink. The red, blue and green circles correspond to the puck locations of shots, dumps and faceoffs respectively.

In the NHL, events are recorded on a play-by-play basis by dedicated statisticians[1]. Additionally, third-party hockey analytics companies provide more in-depth event information, including a greater number of event types and event details, for the NHL and other hockey leagues around the world. Each event is linked with a game-clock timestamp (1-second resolution), and an approximate location where the event occurred on the rink. Generally speaking, the event location corresponds to the approximate location of the puck. Therefore, there exists an expansive knowledgebase of approximate puck location information that has, until now, not been exploited. To this end, this paper explores the following idea: *can we leverage existing hockey event annotations and corresponding broadcast video to predict the location*

[1]Play-by-play event data is publicly available for all NHL games at NHL.com

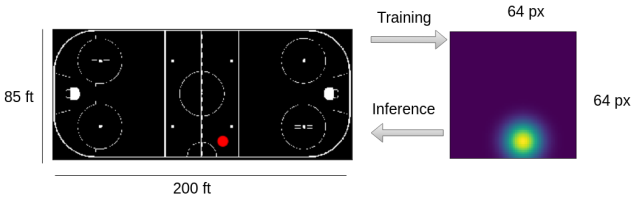

Figure 2: Illustration of the scaling transformation used to transform the puck annotation from the ice rink coordinates to the heatmap coordinates. For training, the annotations are transformed from the ice rink coordinates to the heatmap coordinates, whereas, predicated heatmap is transformed to ice-rink coordinates for inference.

of the puck on the ice?

Using a relatively small dataset of hockey events containing approximate puck locations (distribution shown in Figure 1), we use a 3D CNN to predict the puck position in the rink coordinates using the corresponding 1-second broadcast video clips as input. As such, the 3D CNN is tasked with simultaneously (1) localizing the puck in RGB video and (2) learning the homography between the broadcast camera and the static rink coordinate system. To our best knowledge, this represents a novel computer vision task that shares few similarities with any existing tasks. Drawing inspiration from the domain of human pose estimation, we model the approximate spatial puck location using a 2D Gaussian, as shown in Figure 2.

Background

Pidaparthy and Elder (2019) proposed using a CNN to regress the puck's pixel coordinates from single high-resolution frames collected via a static camera for the purpose of automated hockey videography. Estimating the puck location from a single frame is a challenging task due to the relatively small size of the puck compared to the frame, occlusions from hockey sticks, players, and boards, and the significant motion blur caused by high puck velocities. Furthermore, their method was not based on existing data and thus required extensive data collection and manual annotation.

Remarking that humans can locate the puck position from video with the help of contextual cues and temporal information, our method incorporates temporal information in the form of RGB video to help localize the puck. Additionally, our method differs from Pidaparthy and Elder in that we use puck location information obtained from existing hockey event data, and directly learn the camera-rink homography instead of using a manual calibration.

Methodology

Dataset

The dataset consists of 2716, 60 fps broadcast NHL clips with an original resolution of 1280×720 pixels of one second each with the approximate puck location annotated. The videos are resized to a dimension of 256×256 pixels for computation. The puck locations are evenly distributed throughout the ice rink as can be seen from Figure 1. The dataset is split such that 80% of the data is used for training, 10% for validation and 10% for testing.

Experiment

We use the 18 layer R(2+1)D (Tran et al. 2018) network pretrained on the Kinetics dataset (Kay et al. 2017) as a backbone for regressing the puck location from video. The input to the network consists of 16 video frames $\{I_i \in R^{3\times256\times256} \mid i \in [1,..,16]\}$ sampled from a one second video clip. The 16 frames are sampled from a uniform distribution. For preprocessing, the image frame RBG pixel values are scaled to the $[0, 1]$ range and normalized by the Kinetics dataset mean and standard deviation. The features maps obtained from the 9th layer of the R(2+1)D network is fed into two RegressionBlocks illustrated in Figure 4. The first five layers of the R(2+1)D network are kept frozen during training in order to reduce the computational cost and maintain a batch size of 10 on a single GPU machine. Each regression block consists of a 3D convolutional layer, batch normalization and ReLU non-linearity. The final output of the network is a two-dimensional heatmap $h \in R^{64\times64}$ representing the probability distribution of the puck location. We chose a heatmap based approach instead of directly regressing the puck coordinates in order to account for the uncertainty in the ground truth annotations. The overall network architecture is illustrated in Figure 3 and Table 1. The ground truth heatmap consists of a Gaussian with mean μ equal to the ground truth puck location and standard deviation σ. Mean squared error (MSE) loss between the ground truth and predicted heatmap is minimized during training.

The size of the NHL hockey rink is $200ft \times 85ft$. In order to predict a 64×64 dimensional square heatmap, a scaling transformation $\tau : R^{200\times85} \rightarrow R^{64\times64}$ is applied to the ground truth puck annotations in rink coordinates while training. Let $hmap\_width$ and $hmap\_height$ denote the output heatmap width and height respectively. The transformation matrix is given by:

$$\tau = \begin{pmatrix} \frac{hmap\_width}{200} & 0 & 0 \\ 0 & \frac{hmap\_height}{85} & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

During testing, inverse transformation τ^{-1} is applied to convert back to the rink coordinates. This process is illustrated in Figure 2.

We use the Adam optimizer with an initial learning rate of .0001 with a batch size of 10. We use the Pytorch 1.3 framework on an Nvidia GTX 1080Ti GPU.

Results and Discussion

Accuracy Metric

A test example is considered to be correctly predicted at a tolerance t feet if the L2 distance between the ground truth puck location z and predicted puck location z_0 is less than t feet. That is $||z - z_0||_2 < t$. Let $\phi(t)$ denote the percentage of examples in the test set with correctly predicted position puck position at a tolerance of t. We define the accuracy

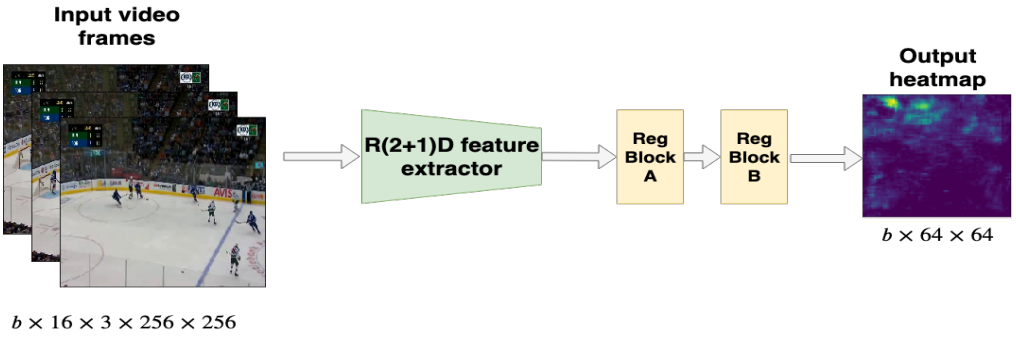

Figure 3: The overall network architecture. Input tensor of dimension $b \times 16 \times 3 \times 256 \times 256$ (b denotes the batch size) is input into the R(2+1)D feature extractor consisting of the first nine layers of the R(2+1)D network. The feature extractor outputs $b \times 8 \times 128 \times 64 \times 64$ tensor representing the intermediate features. The intermediate features are finally input into two regression blocks. The first regression block(Reg Block A) outputs a $b \times 2 \times 32 \times 64 \times 64$ tensor while the second regression block outputs the final predicted heatmap.

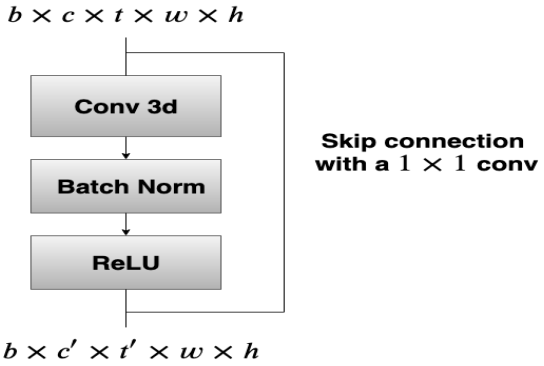

Figure 4: Illustration of the regression block applied after the R(2+1)D network backbone. The input and outputs are 5D tensors where b, c, t, w and h denote batch size, number of channels, temporal dimension, width and height of the feature map respectively. Here $c^{'} < x$ and $t^{'} < t$ since the number of channels and timesteps have to be reduced so that a single heatmap can be generated.

metric as the area under the curve (AUC) $\phi(t)$ at tolerance of $t = 5$ feet to $t = 50$ feet.

Discussion

Figure 5 shows the variation of overall accuracy with tolerance t for the best performing model trained with $\sigma = 25$. The accuracy increases almost linearly reaching $\sim 60\%$ accuracy for $t = 30$ feet. The AUC score for the model is 47.07 %. Figure 6 shows the accuracy vs tolerance plot for the $\sigma = 25$ model, in the horizontal(X) and vertical(Y) directions separately. The model is able to locate the puck position with the highest accuracy in Y(vertical) direction reaching an accuracy of $\sim 65\%$ at a tolerance of $t = 15$ feet. This is because the vertical axis is more or less always visible in the camera field of view. This cannot be said for the

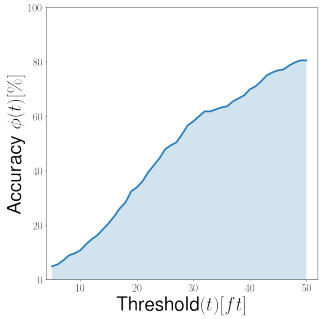

Figure 5: Overall AUC for the best performing model with random sampling and $\sigma = 25$.

Table 1: Network architecture. k,s and p denote kernel dimension, stride and padding respectively. Ch_i and Ch_o and b denote the number of channels going into and out of a block and batch size respectively.

Input $b \times 16 \times 3 \times 256 \times 256$
Feature extractor First 9 layers of R(2+1)D network
RegBlock A Conv3D $Ch_i = 128, Ch_o = 32$ (k = $4 \times 1 \times 1$, s = $4 \times 1 \times 1$, p = 0) Batch Norm 3D ReLU
RegBlock B Conv3D $Ch_i = 32, Ch_o = 1$ (k = $2 \times 1 \times 1$, s = $2 \times 1 \times 1$, p = 0) Batch Norm 3D ReLU
Output $b \times 64 \times 64$

horizontal(X) direction since the camera pans horizontally and hence, the models has to learn the viewpoint changes.

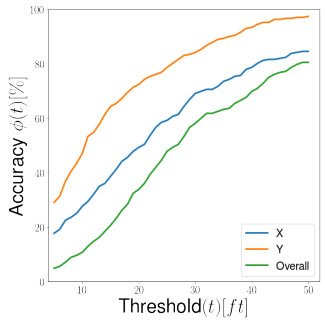

Figure 6: The accuracy curves corresponding to the best performing model.

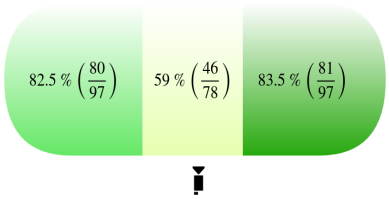

Figure 7: Zone-wise accuracy. The figure represents the hockey rink with the text in each zone represents the percentage of test examples predicted correctly in that zone. The position of the camera is at the bottom.

Table 2: AUC values for different values of σ.

σ	AUC(overall)	AUC(X)	AUC(Y)
10	36.85	48.84	72.25
15	42.51	53.86	**77.12**
20	45.80	57.31	76.66
25	**47.07**	**58.85**	76.78
30	42.86	54.23	76.76

Table 2 shows the variation of AUC for different values of σ. The highest AUC score achieved is corresponding to $\sigma = 25$ (47.07 %). A lower value of σ results in a lower accuracy. A reason for this can be that with lower σ, the ground truth Gaussian distribution becomes more rigid/peaked, which makes learning difficult. For a value of $\sigma > 25$, the accuracy again lowers because the ground truth Gaussian becomes very spread out, which lowers accuracy on lower tolerance levels.

Two kinds of sampling techniques were investigated: 1) Random sampling from a uniform distribution 2) Constant interval sampling at an interval of 4 frames. Random sampling outperforms uniform sampling because it acts as a form of data augmentation. This is shown in Table 3.

Figure 8 shows the zone-wise accuracy of the model. A prediction is labelled as correct if it lies in the same zone as the ground truth. The model shows good performance in the offensive and defensive zones with an accuracy greater than 80%. The model maintains reasonable performance when the defensive and offensive zones are further split into two (Figure 8).

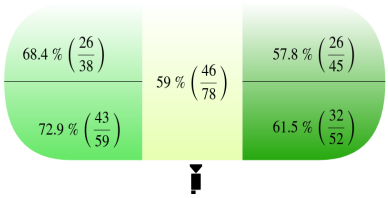

Figure 8: Zone-wise accuracy with offensive and defensive zones further split into two. The figure represents the hockey rink with the text in each zone represents the percentage of test examples predicted correctly in that zone. The position of the camera is at the bottom.

Table 3: Comparison between uniform and random sampling settings. Random sampling outperforms uniform sampling because it acts as a form of data augmentation.

Sampling	σ	AUC(overall)	AUC(X)	AUC(Y)
Random	20	45.80	57.31	76.66
Constant interval	20	36.55	49.24	71.41

Conclusion and Future Work

We have presented a novel method to locate the approximate puck position from video. The model can be used to know the zone in which the puck was present at a particular moment in time, which can be of practical significance to know the exact location of play and as a prior information for recognizing game events. The results obtained are preliminary and in the future more cues such as player detections, player trajectories on ice and optical flow can be taken into account to obtain more accurate results. It would also be interesting to apply the proposed methodology in sports such as soccer.

Acknowledgment

This work was supported by Stathletes through the Mitacs Accelerate Program and Natural Sciences and Engineering Research Council of Canada (NSERC).

References

Cavallaro, R. 1997. The FoxTrax hockey puck tracking system. *IEEE Computer Graphics and Applications* 17(2):6–12.

Fani, M.; Neher, H.; Clausi, D. A.; Wong, A.; and Zelek, J. 2017. Hockey action recognition via integrated stacked hourglass network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 29–37.

Gulitti, T. 2019. NHL plans to deploy Puck and Player Tracking technology next season.

IIHF. 2018. Survey of Players. Available online: https://www.iihf.com/en/static/5324/survey-of-players.

Kay, W.; Carreira, J.; Simonyan, K.; Zhang, B.; Hillier, C.; Vijayanarasimhan, S.; Viola, F.; Green, T.; Back, T.; Natsev, A.; Suleyman, M.; and Zisserman, A. 2017. The kinetics human action video dataset. *ArXiv* abs/1705.06950.

Lu, W.-L.; Okuma, K.; and Little, J. J. 2009. Tracking and recognizing actions of multiple hockey players using the boosted particle filter. *Image and Vision Computing* 27(1-2):189–205.

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

Thomas, A. C. 2006. The impact of puck possession and location on ice hockey strategy. *Journal of Quantitative Analysis in Sports* 2(1).

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

Tran, D.; Wang, H.; Torresani, L.; Ray, J.; LeCun, Y.; and Paluri, M. 2018. A closer look at spatiotemporal convolutions for action recognition. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 6450–6459.