Hockey Action Recognition via Integrated Stacked Hourglass Network

Mehrnaz Fani
Shiraz University
fani.mehrnaz@shirazu.ac.ir

Helmut Neher
University of Waterloo
hneher@uwaterloo.ca

David A. Clausi , Alexander Wong, John Zelek
University of Waterloo
[dclausi@uwaterloo.ca, a28wong@uwaterloo.ca, jzelek@uwaterloo.ca]

Abstract

A convolutional neural network (CNN) has been designed to interpret player actions in ice hockey video. The hourglass network is employed as the base to generate player pose estimation and layers are added to this network to produce action recognition. As such, the unified architecture is referred to as action recognition hourglass network, or ARHN. ARHN has three components. The first component is the latent pose estimator, the second transforms latent features to a common frame of reference, and the third performs action recognition. Since no benchmark dataset for pose estimation or action recognition is available for hockey players, we generate such an annotated dataset. Experimental results show action recognition accuracy of 65% for four types of actions in hockey. When similar poses are merged to three and two classes, the accuracy rate increases to 71% and 78%, proving the efficacy of the methodology for automated action recognition in hockey.

1. Introduction

Action recognition in computer vision is an important and popular problem in the application of analyzing sport videos. Action recognition provides a benefit to coaches, analysts and spectators by providing content for coaches and analysts to evaluate player performance and for spectators to view content. Ice hockey is one example of a sport with an application of action recognition. Although only a limited amount of action recognition or even computer vision research has been done in the field of hockey, action recognition can be applied to hockey analytics to analyze characteristics of hockey players and teams. Current methods in hockey analytics utilize manually assessed statistics to evaluate player performance, yet, practitioners want more information using less time consuming methods. Hockey player pose and hockey action recognition are valuable pieces of information that potentially can help coaches in assessing player performance. The focus of this research is to perform action recognition using latent pose in hockey videos and images.

Pose estimation and action recognition are challenging problems in hockey which can be scaled to other types of sports. Action recognition challenges specific to hockey include bulky clothing that deforms a player's body-shape, a team's jersey (white) that is highly similar to the background (the ice and boards), equipment (padding) that tends to occlude joints and limbs, and a high speed of movement due to skating ability that leads to blurring. These challenges make automated action recognition and pose estimation in hockey quite difficult.

In this article videos captured by a single camera is employed, and a convolutional neural network (CNN), called Action recognition Hourglass Network (ARHN), is introduced that extracts pose features from hockey images and/videos and utilizes them for action recognition. Although the use of depth sensors can be employed (see the background found in Section 2), that method is expensive and the data gathered is too noisy. Action recognition for broadcast videos, although more challenging, is more desirable and more realistic.

A dataset of annotated hockey images is generated; to the best of our knowledge, there is no publicly available benchmark hockey dataset for action recognition and pose estimation. In this dataset, video frames of hockey players performing four types of activities (namely, cross-overs, straight skating, per-shot, and post-shot) are labeled and body joint locations are annotated.

The main contributions include the following: 1) introducing a framework for action recognition in videos (Section 3), 2) proposing the ARHN architecture as a unified deep structure for action recognition (Section 3), 3) creation of an action recognition dataset using frames from ice hockey video (Section 4), and 4) successful application of

the ARHN to the generated dataset for automatic recognition of hockey player actions (Section 4).

This research focuses on utilizing pose information for action recognition and does not employ temporal features for two reasons. First, a player's pose, as a static feature, is a strong clue for action recognition, and second incorporating temporal information like motion descriptors arises the need for a much bigger dataset to be used for training a deep structure that incorporates temporal information.

2. Background

The background section is composed of two subsections: current pose-based action recognition techniques used in sports and computer vision research applied to ice hockey.

2.1. Pose-based Action Recognition

Many works in action recognition use dense trajectory features including HOG, HOF, and MBH [8, 17, 19, 14, 4] in addition to pose estimation. Pishchulin *et al*. [17] explore combinations of dense trajectories and pose estimation noting that combinations may improve accuracy of action recognition given that the pose estimates are unable to accurately label the pose of a person. Jhuang et. al [19] compares dense trajectory, a low/mid-level method, separately against pose estimation, a high-level method, determining that methods incorporating pose features outperform low/mid level feature methods.

One method to incoporate dense trajectories and pose estimation is using AND-OR graph models [8, 14]. One implementation incorporates motion, geometry of joints (pose) and appearance [8]. The model uses HOF/HOG for motion appearance as a part node. The pose node has a projected 3D view and then it is placed into 'different' view nodes (difference viewpoints); this approach is tested using 2D video input and is to help evaluate actions in various viewpoints. Another model incorporates poselets in addition to HOG/HOF within the And-Or graph model [14].

Similarily to incorporating poselets, Desai *et al*. [6] present an approach based on combining 3 compositional models (i.e., poselets, visual phrases, and pictorial structure models) for modeling human pose and interacting objects. Phraselets are introduced and employed in a Flexible Mixture of Parts (FMP) framework to capture relations between parts and a separate compositional model per action class is defined. Output of the model are action labels, articulated human pose, object pose, and occlusion flags. Phraselets, like most of the methods in action recognition, are designed for recognizing coarse actions that are quite different in nature (like horse ridding verses taking photo), not for fine action recognition (e.g., discriminating between 2 different movements of a hockey player).

Iqbal and Gall [18] introduce a method for repeatedly altering between pose estimation and action recognition. They adopt standard pictorial structure model (PS model) for human pose estimation and condition it on action types to do efficient inference. Starting with uniform prior on all action classes, the pose in each frame is predicted, and by using the estimated poses, the probabilities of the actions are estimated.

Recently, deep structures have dominated most of previous descriptors and models for pose estimation and are giving promising results in action recognition.

In Chéron *et al*. [4], a pose based CNN providing a descriptor for action recognition task is introduced. Pose estimation is performed using a method given by Cherian *et al*. [3], and is utilized for determining four different regions or body parts in images. Next, optical flow and raw image pixels over patches of body parts are given to separate CNNs to generate motion and appearance descriptors for each frame. Descriptors per frame, and their consecutive differences in successive frames, are aggregated by max and min pooling over time and normalized to generate static and dynamic video descriptors, which are concatenated to form P-CNN descriptor. Beside P-CNN descriptor, three different Improved Dense Trajectory features (i.e., HOG, HOF,and MBH) with Fisher Vector coding are also computed. Action recognition is performed using a linear SVM over P-CNN descriptors and IDT features. This method as explained does not use the pose estimation directly as a feature but rather employs it for determining the region of interest for patch selection from images, while we believe pose information, if determined precisely, is intrinsically a strong clue for action recognition.

Similar research in action recognition in sports uses pose estimation as a latent variable in a unified action recognition in still images [20]. Like Yang *et al*. [20], the authors seek to unify pose estimation and action recognition to improve action recognition performance.

This literature review shows pose can be used as a strong feature for action recognition and employing power architectures such as deep networks will increase the accuracy of pose-based action recognition. Therefore, in this work a pose-based deep network incoporating latent pose estimation is implemented for action recognition.

2.2. Computer Vision and Action Recognition in Hockey

Within the sport of hockey, computer vision research has been limited to tracking [2, 16, 15, 9, 11] , rectification of broadcast hockey video [7], crowd analysis [5] providing a hockey crowd dataset, and very few results in action recognition [11, 10, 12].

In the three papers by Lu *et al*. [11, 10, 12], HOG descriptors are used with various training methods such as support vector machines, prior information is extracted from videos and sequences of images are implemented as

input for action recognition. The mentioned papers, however, do not describe methods for extracting higher level features such as pose. The activities evaluated from the aforementioned papers are based on actions of skating such as skating left, skate right, skate in, skate out, skate left 45, and skate right 45 rather than other action of hockey that focus on the whole body.

This summary represents the limited extent of the published research in the field of computer vision applied to ice hockey. In this work, a significant state-of-the-art contribution by developing a methodology to automatically determine actions of a hockey player based on latent pose estimation derived from video frames is presented.

3. Methodology

3.1. Overview

As indicated earlier, this research involves the development, implementation, and testing of a new method to perform action recognition. This method is applied to recognizing actions of ice hockey players as an example. The ARHN uses features based on latent pose estimation to estimate action recognition using single video frames of hockey players. An overview of the framework is demonstrated in Fig. 1 and is described in Section 3.2.

3.2. Proposed General Framework

Proposed action recognition framework in video, is illustrated in Fig.1. As shown in Fig. 1, a hockey video-segment is converted to a sequence of frames. In each frame a player is tracked, and a coordinate of his body center is determined. Next, the frame resolution is adjusted to the proper input size (i.e., 720×1280) of the network. Then a region of interest (with size 250×250), centered at a player's body-center is cropped from the image and is given to the ARHN network. The network, by finding heatmaps (where each heatmap corresponds to the predicted probability of a joint's presence at each image-pixel [13]), generates the pose estimation which is then used to estimate player action. Four different types of hockey player actions are considered which are: cross-over, straight skating, pre-shot, and post-shot. Details of the ARHN structure are presented and discussed in the next subsection.

3.3. Network Architecture

The general structure of the ARHN is presented in Fig. 2 and broken into three components. The first component is the stacked hourglass network ([13] which inputs the raw image and generates a set of heatmaps that defines the pose, as the latent feature. The second component of the network is the latent feature transformer that receives the latent features and transforms them to a common frame of reference. The third component is the action recognition

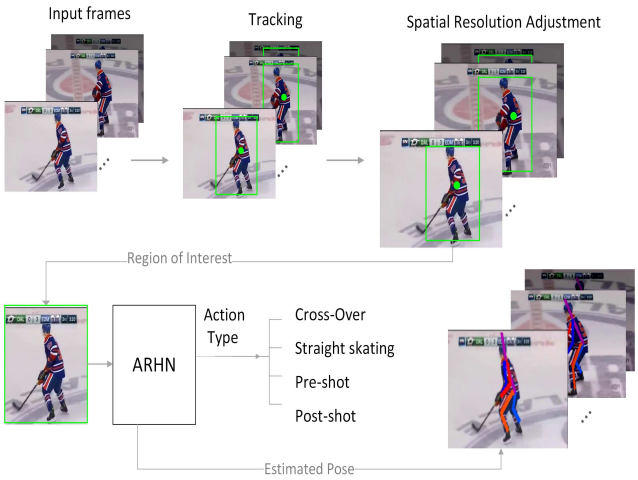


Figure 1: Implemented framework for hockey action recognition through pose estimation for hockey images/video frames. The framework begins by extracting video frames as input, the body-center of the player is determined using tracking means, then the image size is scaled and fed into the network. The network then classifies the action and overlays the estimated pose on the image.

classifier which is composed of six fully-connected layers and classifies a hockey player' s action type. Sequencing these three parts, as shown in Fig.2, constructs the ARHN network as a unified deep structure.

The pose estimator component, implicitly learns the pose of a hockey player through the use of a generated set of statistical probability heatmaps that identify the joint locations of a hockey player in a still image. Then the latent feature transformer, scales and shifts the learned pose, forming a feature vector. The fully connected layers in the third component perform the action recognition task.

In order to understand the ARHN, a brief overview of the original hourglass network, in conjunction with a description of latent feature transformer and fully-connected layers are provided respectively in subsections 3.4, 3.5, and 3.6.

3.4. Latent Pose Estimation via Stacked Hourglass Network

The stacked hourglass network is a deep convolutional network architecture composed of multiple hourglass modules put together in series [13]. Each hourglass module has convolutional, max-pooling, and up-sampling layers as its basic elements to realize a bottom-up, top-down mechanism for generating feature maps. In bottom-up sequence, successive convolution and max pooling layers are engaged to bring the resolution of feature maps to 4x4 pixels. In

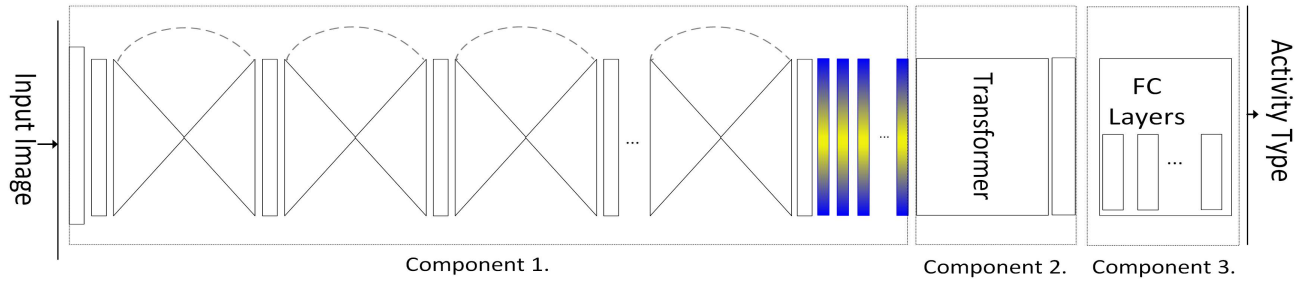


Figure 2: Proposed ARHN for action recognition identifying the three components. Component 1: pose estimation using an hourglass network. Component 2: feature transformation to transform poses into a common frame of reference Component 3: action recognition represented by fully connected layers.

Figure 3: Statistical heat maps demonstrating the probability of the location of the right ankle, right knee, and right hip (left to right) for a hockey player.

the top-down sequence, feature maps are up-sampled using nearest neighbor. The major elements of this architecture are skip connections between bottom-up and top-down sections of the hourglass, which are shown by dashed arches in Fig.2. These skip connections preserve information of high resolution feature maps, in the first section of network, to be combined with features of other scales, in the second section. The hourglass network generates a set of 16 statistical heatmaps. Fig.3 provides instances of some heat maps for the right ankle, right knee and right hip of a hockey player. These heatmaps actually form the latent pose features for the ARHN network.

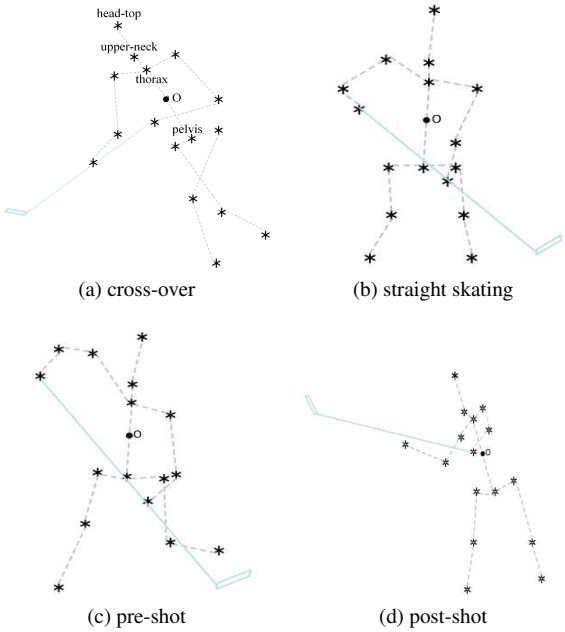


Figure 4: Typical poses for 4 different actions of a hockey player

3.5. Latent Feature Transformer

The second component is a feature transformer that transforms pose heatmaps to a common frame of reference by performing spatial translation and scaling in 2-D plane. The location of a peak in a heatmap gives the predicted coordinate of a joint for the input image. A specific constellation of joints (i.e., geometrical arrangement of a set of joints) shows the pose of a player. A player's pose potentially should represent a particular type of action, that is performed; typical poses for four types of actions in hockey are indicated in Fig. 4. However, poses that represent the same action type can vary significantly in the joint position, orientation, and sizes. To generate a more consistent representation for poses, referred to as canonical poses, the feature transformer is used. This component generates the canonical poses from the heatmaps to generate a better pose representation to be used as input into the action recognition component.

The latent feature transformer, is demonstrated in Fig.5(a). All joint coordinates are shifted with respect to a point defined as the body center (x_0, y_0), namely, the point

halfway between the thorax and pelvis keypoints indicated by "O" on the stickmen in Fig. 4. Joint coordinates are scaled by scaling ratio S as per Eq. (1). S is the ratio of the average head size of players in all training images (N) and H_n is the head size of the player in the n^{th} image. Head size is the distance between the "head top" and "upper neck" keypoints (Fig. 4).

$$S = \frac{\sum_{n=1}^{N} H_n}{H_n} \tag{1}$$

As shown in Fig. 5, besides transformed coordinates (i.e., $[x_i, y_i]^T$) of 16 body-joints, angles (α_j) between some joints are also calculated. Angles which are computed are between (right & left)- shoulder, (right & left)-elbow, (right & left)-hip, and (right & left)- knee joints. The output of the latent feature transformer is a 40-dimensional vector named the canonical pose (p_c) given in Eq. (2). This vector is formed by concatenation of joint angles and transformed keypoint coordinates. The canonical pose is the feature that is next evaluated by the third component of the ARHN to perform action recognition.

$$p_c = \begin{bmatrix} \alpha & X & Y \end{bmatrix} \tag{2}$$

$$X = \begin{bmatrix} x_1, & x_2, & ..., & x_i, & ... & x_{16} \end{bmatrix}$$

$$, Y = \begin{bmatrix} y_1, & y_2, & ..., & y_i, & ... & y_{16} \end{bmatrix}$$

$$, \alpha = \begin{bmatrix} \alpha_1, & \alpha_2, & ..., & \alpha_j, & ... & \alpha_8 \end{bmatrix}$$

3.6. Action Recognition Component

The last component of the network is illustrated in Fig. 6. This component is composed of six fully connected layers to recognize activities. The fully connected layers receive the 40-dimensional feature vector from the latent feature transformer, passing it through five fully connected layers with sigmoid activation functions and a final layer of four neurons with a hard-limit function to recognize one of the four types of activities for the input image. The number of neurons in each layer is indicated in Fig. 6. Note that the number of layers and the number of neurons per layer are determined empirically.

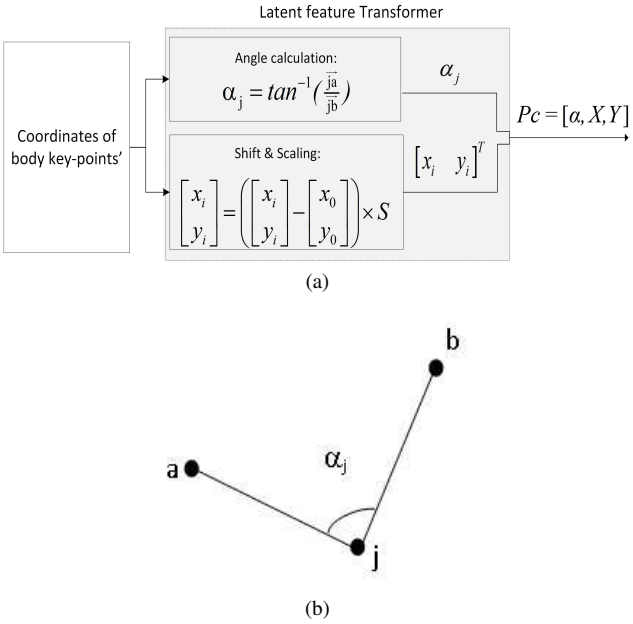


Figure 5: (a) Latent feature transformer, which generates canonical pose vector by shifting and scaling joint coordinates and computing the joint angles. (b) Angle of joint "j" (i.e., α_j) is the smaller angle between vectors ja and jb.

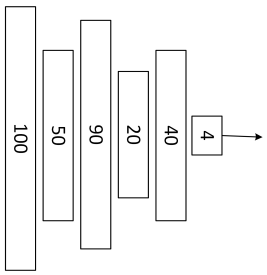


Figure 6: Action recognition component consisting of 6 fully connected layers beginning with a layer of 100 neurons to 50, 90, 20, 40 and ending with a fully connected layer of 4 to perform action recognition.

4. Testing and Results

Experiments that are conducted here assess the performance of ARHN for action recognition in the context of hockey. Both visual and numerical evaluations are provided.

4.1. Dataset Preparation

In machine learning, and particularly deep learning problems, having access to a proper dataset is a crucial requirement. Deep networks generally use supervised or semi-supervised algorithms for learning, which heavily rely on annotated data during training. Deep networks are designed to extract information from raw input data, therefore, performance relies on the data samples presented to deep networks. If the provided data are not representative of the problem, or the number of training samples is limited, the machine learning method fails to properly tune its parameters and cannot provide an accurate model for solving the problem.

In the context of hockey, no standard set of annotated hockey images for pose estimation or action recognition is available. Therefore, in this work, a dataset, named

#	Key-point	#	Key-point
1	Right ankle	10	Head top
2	Right knee	11	Right wrist
3	Right hip	12	Right elbow
4	Left hip	13	Right shoulder
5	Left knee	14	Left shoulder
6	Left ankle	15	Left elbow
7	Pelvis	16	Left wrist
8	Thorax	17	top of stick
9	Upper neck	18	end of stick

Table 1: List of annotated key-points for each frame.

HARPE, has been collected and annotated for this purpose.

- Video segments are captured from a set of hockey videos and converted to video frames.
- Video frames are categorized into classes based on the four hockey actions: cross-overs, straight skating, pre-shot, and post-shot.
- Very low quality frames, and frames unrepresentative of classes, are manually detected and discarded.
- Spatial resolution of each frame is adjusted to the proper size for delivering to the network i.e., 720×1280.
- A hockey player is tracked in all frames to determine his body center in pixel coordinates.
- For each frame, the positions of 16 body joints (Table 1) for the player of interest is annotated and the action type is labeled.
- The two ends of the hockey sticks are also annotated in each frame for future use.

In summary, keypoints are annotated in 887 frames with an associated action label. The dataset has 1676 frames of cross-overs, 271 frames of straight skating, 245 frames of pre-shooting, and 203 frames of post-shooting. We are planning to make the dataset publicly available.

4.2. Accuracy of Action Recognition

To evaluate the accuracy of action recognition the images are randomly divided into three groups: 70% training, 15% validation, and 15% validation. Training images are passed through the ARHN network and the parameters of network are tuned accordingly. Due to the limited size of the provided data, parameters of hourglass layers are hardly affected (weights of hourglass layer are pre-trained by general human poses on MPII dataset [1]), while parameters

Class #	1	2	3	4
Precision (%)	68.3	7.18	75.9	79.5
Recall (%)	68.6	74.1	77.0	73.0

Table 2: Performance of ARHN for training.

Class #	1	2	3	4
Precision (%)	64.5	68.8	72.6	64.1
Recall (%)	69.5	68.9	68.4	64.1

Table 3: Performance of ARHN for validation.

Class #	1	2	3	4
Precision (%)	61.7	67.0	68.3	63.1
Recall (%)	61.7	67.0	68.1	63.1

Table 4: Performance of ARHN for testing.

of the fully connected layers are the ones that are mainly learned during the training phase. This process has been repeated with fifteen randomly selected groups, and the average performance of ARHN network for action recognition is reported. The 70/15/15 splitting of data and averaging over 15 runs, is validated in subsection 4.3.

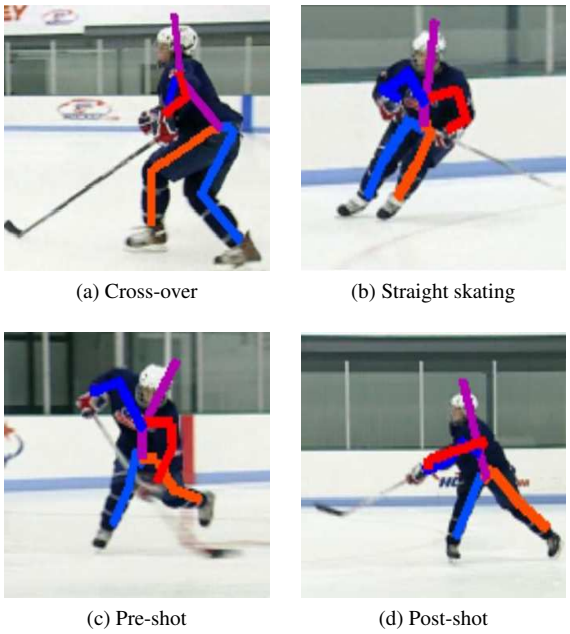
For this purpose, precision and recall rates for training, testing, and validation images are computed and provided respectively in Tables 2, 3, and 4 for each of the four class types; where 1 represents cross-overs, 2 represents straight skating, 3 represents pre-shot, and 4 represents post-shot.

The precision and recall rates of Table 4, for the test data, show that the network has precision of about 65% for each class. However, in many cases, a hockey players' pose in cross-over and straight skating (the two first classes) are quite similar to each other. It is also the case for pre-shot and post-shot (the two last classes). For each type of action, some examples of correctly classified and misclassified images are illustrated in Figs. 7 and 8. In Fig. 7, all images follow the typical action poses shown in Fig. 4, so they are all correctly classified by the ARHN. In contrast, images of Fig. 8 are all misclassified because they deviate from their true class and mimic a different class. Considering player poses, misclassification of Fig. 8 by ARHN can be justified. This subject is further investigated in the next experiment.

4.3. Effect of Merging Classes

The purpose of this experiment is to show that by merging similar classes, accuracy of classification can be improved. In Fig. 9, a confusion matrix for one run on training data is provided.

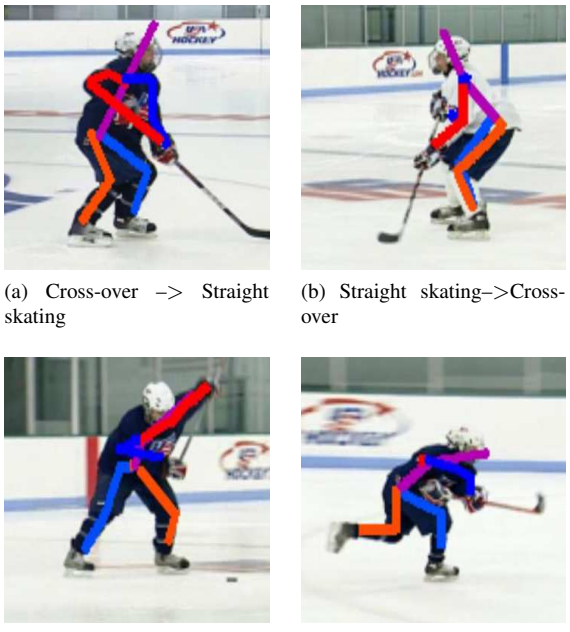
The confusion matrix in Fig. 9 shows that most mis-



(a) Cross-over (b) Straight skating

(c) Pre-shot (d) Post-shot

Figure 7: Examples of activities, correctly classified.



(a) Cross-over –> Straight skating (b) Straight skating–>Cross-over

(c) Pre-shot–>Post-shot (d) Post-shot–>Pre-shot

Figure 8: Examples of missclassified activities. In each case the true class-type followed by the predicted class-type are shown under the image in question.

classifications occur between classes 3 and 4 (pre-shot and post-shot), as well as classes 1 and 2 (cross-over and straight skating); shooting classes are clearly distinct from the skating classes. Therefore, in Table 5, the effect of merging similar classes on accuracy of action recognition is investigated. Mean classification accuracy averaged over 15 and then 1000 runs are reported for three different testing conditions.

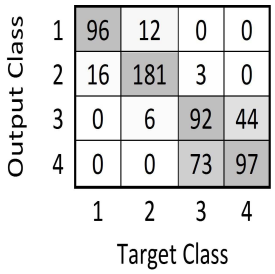


Figure 9: Confusion matrix of action recognition for one run.

Class Indices	1,2,3,4	1,2,(3,4)	(1,2),(3,4)
Mean 15 runs(%)	65.14	71.13	78.32
Mean 1000 runs(%)	65.47	69.08	78.49
Variance 1000 runs	0.0064	0.0043	0.0030

Table 5: Accuracy of action recognition over 15 and 1000 runs for three testing conditions: evaluating classes 1,2,3 and 4 as separate classes, evaluating class 1 and 2 separately with classes 3 and 4 as a single class, and evaluating classes 1 and 2 as one separate class and classes 3 and 4 as another class.

In the first test none of the classes are merged together. In the second test the two last classes (i.e., pre-shot and post-shot) are merged together. Finally, in the third test the first two classes (i.e., cross-over and straight skating) are also combined. Accuracy of recognition for each of these testing conditions are provided in the three columns of Table 5. Table 5 demonstrates that by unifying similar classes, the mean accuracy increases. Also, Table 5 shows that the mean accuracy over 15 runs is close to the mean accuracy over 1000 runs. The low variance over 1000 runs validates that fewer runs (e.g., 15) should be sufficient for representative results. The result of this test for 1000 runs are also demonstrated in the form of histograms in Fig. 10 for each of the three testing conditions. The histograms show that by merging the classes, mean accuracy increases and variance decreases resulting in the histogram to look more concentrated.

5. Conclusion

In this article, a deep structure called ARHN network is designed and implemented that successfully performs ac-

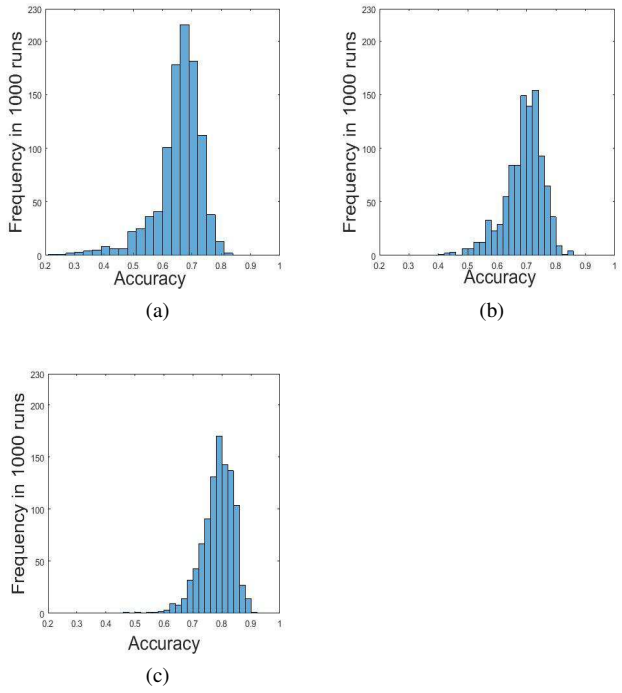


Figure 10: Histograms of accuracy over 1000 runs for random selection of test samples. (a) all 4 classes as separate classes (b) classes 3 and 4 acting as one class and class 1 and 2 as separate classes and (c) classes 1 and 2 are a single class as well as classes 3 and 4.

tion recognition in the sport of hockey using latent pose estimation features. A labeled dataset of hockey poses and of four action classes have also been introduced as a bench mark dataset for action recognition in hockey. Body joint locations in all images of this dataset are annotated and can be used as ground-truth for pose estimation.

Hockey analytics derived from computer vision methods is in its infancy. So, this work could help the coaches and hockey analysts to evaluate player performance from a more scientific viewpoint. Note that pose estimation in a hockey game is extremely challenging due to occlusions caused by protective equipment, high level of motion blur due to the speed of the game, and a high degree of player interactions caught in a standard camera view.

In the collected dataset, images of goalies are excluded, because of their inconsistency in clothing with other players'. Therefore,in feature works a separate network for goalies could be trained. Applying the proposed method to goalies would require a complete adjustment of the pose estimation to account for goalie pads. This is far more work especially with respect to preparing training data for the hourglass network.

Moreover, architecture of ARHN network could be changed to include the hockey stick key-points and improve the accuracy of action recognition.

Acknowledgments

This work is partly funded by the Natural Sciences and Engineering Research Council of Canada (NSERC).

References

[1] M. Andriluka, L. Pishchulin, P. Gehler, and B. Schiele. 2d human pose estimation: New benchmark and state of the art analysis. In *2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Columbus, USA, June 2014.

[2] Y. Cai. Robust visual tracking for multiple targets. In *2006 Proceedings of the European Conference on Computer Vision (ECCV)*, pages 107–118, Graz,Austria, 2006.

[3] A. Cherian, J. Mairal, K. Alahari, and C. Schmid. Mixing body-part sequences for human pose estimation. In *2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2361–2368, Columbus, USA, June 2014.

[4] G. Chéron, I. Laptev, and C. Schmid. P-cnn: Pose-based cnn features for action recognition. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 3218–3226, Santiago, Chile, 2015.

[5] D. Conigliaro, P. Rota, F. Setti, C. Bassetti, N. Conci, N. Sebe, and M. Cristani. The S-HOCK dataset: Analyzing crowds at the stadium. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2039–2047, Boston, USA, June 2015.

[6] C. Desai and D. Ramanan. Detecting actions, poses, and objects with relational phraselets. In *Proceedings of the 12th European Conference on Computer Vision (ECCV) - Volume Part IV*, ECCV'12, Florence, Italy, 2012.

[7] A. Gupta, J. J. Little, and R. J. Woodham. Using line and ellipse features for rectification of broadcast hockey video. In *The 14th Canadian Conference on Computer and Robot Vision (CRV'15)*, pages 32–39, Halifax, Canada, 2011. IEEE Computer Society.

[8] H. Jhuang, J. Gall, S. Zuffi, C. Schmid, and M. J. Black. Towards understanding action recognition. In *2013 IEEE International Conference on Computer Vision (ICCV)*, pages 3192–3199, Sydney, Australia, Dec 2013.

[9] F. Li and R. J. Woodham. Video analysis of hockey play in selected game situations. *Image and Vision Computing*, 27(12):45 – 58, 2009.

[10] W.-L. 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, Quebec, Canada, June 2006.

[11] W. L. Lu and J. J. Little. Tracking and recognizing actions at a distance. In *Proceedings of the ECCV Workshop on Computer Vision Based Analysis in Sport Environments (CVBASE '06)*, Graz, Austria, May 2006.

[12] W. L. 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(1–2):189–205, 2009.

[13] A. Newell, K. Yang, and J. Deng. Stacked hourglass networks for human pose estimation. In B. Leibe, J. Matas, N. Sebe, and M. Welling, editors, *2016 Proceedings of the European Conference on Computer Vision (ECCV)*, pages 483–499, Amsterdam, Netherlands, October 2016. Springer International Publishing.

[14] B. X. Nie, C. Xiong, and S. C. Zhu. Joint action recognition and pose estimation from video. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1293–1301, Boston, USA, June 2015.

[15] K. Okuma, D. G. Lowe, and J. J. Little. Self-learning for player localization in sports video. *Computing Research Repository*, abs/1307.7198, 2013.

[16] K. Okuma, A. Taleghani, N. D. Freitas, O. D. Freitas, J. J. Little, and D. G. Lowe. A boosted particle filter: Multitarget detection and tracking. In *2004 Proceedings of the European Conference on Computer Vision (ECCV)*, pages 28–39, Prague, Czech Republic, 2004.

[17] L. Pishchulin, M. Andriluka, and B. Schiele. Fine-grained activity recognition with holistic and pose based features. In *2014 German Conference on Pattern Recognition (GCPR)*, pages 678–689, Münster, Germany, 2014. Springer International Publishing.

[18] M. G. U. Iqbal and J. Gall. Pose for action - action for pose. In *Proceedings of the 12th IEEE International Conference on Automatic Face and Gesture Recognition (FG)*, Washington, DC, USA, May 2017.

[19] J. Wang, X. Nie, Y. Xia, Y. Wu, and S. C. Zhu. Cross-view action modeling, learning, and recognition. In *2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2649–2656, June 2014.

[20] W. Yang, Y. Wang, and G. Mori. Recognizing human actions from still images with latent poses. In *2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2030–2037, San Francisco, USA, June 2010.