Estimations of 3D velocities from a single camera view in ice hockey

BEATRICE BJERING
Estimations of 3D velocities from a single camera view in ice hockey

Beräkningar av 3D hastigheter från en kameravinkel i ishockey

BEATRICE BJERING

Degree Project in Medical Engineering Stockholm, Sweden 2019
Supervisor: Jonas Wåhlsén and Qiantailang Yuan
Reviewer: Svein Kleiven
Examiner: Mats Nilsson
School of Engineering Sciences in Chemistry, Biotechnology and Health
Abstract

Ice hockey is a contact sport with a high risk of brain injuries such as concussions. This is a serious health concern and there is a need of better understanding of the relationship between the kinematics of the head and concussions. The velocity and the direction of impact are factors that might affect the severity of the concussions. Therefore the understanding of concussions can be improved by extracting velocities from video analysis.

In this thesis a prototype to extract 3D velocities from one single camera view was developed by using target tracking algorithms and homography. A validation of the method was done where the mean error was estimated to 21.7%. The prototype evaluated 60 cases of tackles where 30 resulted in concussions and the other 30 tackles did not result in concussions. No significant difference in the velocities between the two groups could be found. The mean velocity for the tackles that resulted in concussions were 6.55 m/s for the attacking player and 4.59 m/s for the injured player. The prototype was also compared with velocities extracted through SkillSpector from a previous bachelor thesis. There was a significant difference between the velocities compiled with SkillSpector and the developed prototype in this thesis. A validation of SkillSpector was also made, which showed that it had a mean error of 37.4%.

Keywords: Video analysis, Target tracking, Homography, Concussions, Ice hockey.
**Sammanfattning**


I denna rapport utvecklades en prototyp för att ta fram 3D hastigheter från en kameravinkel genom att använda målsökningsalgoritmer och homografi. En validering av prototypen gjordes där medelfelet uppskattades till 21.7%. Prototypen utvärderade även 60 fall av tacklingar där 30 resulterade hjärnskakningar och där de andra 30 tacklingarna inte resulterade i hjärnskakningar. Ingen signifikant skillnad mellan de två grupperna kunde påvisas. Medelhastigheten för tacklingarna som resulterade i hjärnskakning var 6.55 m/s för den attackerande spelaren och 4.59 m/s för den skadade spelaren. Prototypen jämfördes också med hastigheter som tagits fram med SkillSpector i ett tidigare kandidatexamensarbete. Det var en signifikant skillnad mellan de hastigheter som togs fram med prototypen och de som tog fram med SkillSpector. En validering av SkillSpector gjordes också, som visade att medelfelet var 37.4%.

**Nyckelord:** Videoanalys, Målsökning, Homografi, Hjärnskakningar, Ishockey.
Acknowledgements

To begin with, I would like to thank my reviewer Svein Kleiven for helping me find this project. I would also like to thank my supervisors Jonas Wåhlsén and Qiantailang Yuan for the support and valuable feedback during the process.

I would also like to thank my seminar group Jia Cheng, Ekant, Sina and Steinunn and our group supervisor Xiaogai Li for the useful comments and reviews on my report, helping me to improve the report in the best way possible.

Finally I would like to thank my family and my boyfriend for always encouraging and supporting me. Thank you for believing in me.
<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TBI</td>
<td>Traumatic brain injury</td>
</tr>
<tr>
<td>SHL</td>
<td>Swedish hockey league</td>
</tr>
<tr>
<td>NHL</td>
<td>National hockey league</td>
</tr>
<tr>
<td>IIHF</td>
<td>International ice hockey federation</td>
</tr>
<tr>
<td>DLT</td>
<td>Direct linear transformation</td>
</tr>
<tr>
<td>FAST</td>
<td>Features from accelerated segment test</td>
</tr>
<tr>
<td>BRISK</td>
<td>Binary robust invariant scalable keypoint</td>
</tr>
<tr>
<td>KLT</td>
<td>Kanade-Lucas-Tomasi</td>
</tr>
<tr>
<td>CAMShift</td>
<td>Consciously adapted mean shift</td>
</tr>
<tr>
<td>COCO</td>
<td>Common objects in context</td>
</tr>
<tr>
<td>CNNs</td>
<td>Convolutional neural networks</td>
</tr>
<tr>
<td>LMS</td>
<td>Least median of squares</td>
</tr>
<tr>
<td>RANSAC</td>
<td>Random sample consensus</td>
</tr>
</tbody>
</table>
# Contents

1 Introduction ........................................... 1

2 Aim and objectives ................................... 2

3 Method .................................................. 3
   3.1 Choice of algorithm for insertion of object points .... 3
   3.2 Choice of tracking algorithm ............................ 4
   3.3 Choice of depth estimation .............................. 5
      3.3.1 Rink dimensions .................................. 6
   3.4 Velocity estimation ..................................... 8
   3.5 Prototype ........................................... 9
   3.6 Validation of the prototype ............................ 11
   3.7 Validation of SkillSpector ............................. 12
   3.8 Video collection ...................................... 13

4 Results .................................................. 14
   4.1 Velocities in SkillSpector compared with the prototype . 14
   4.2 Velocities for 60 tackles compiled with the prototype ... 15
   4.3 Validation of the prototype ............................ 19
   4.4 Validation of SkillSpector ............................. 20

5 Discussion ............................................. 21
   5.1 Performance and sources of error of the prototype .... 21
   5.2 Evaluation of SkillSpector ............................ 22
   5.3 Velocities estimated by the prototype ................. 22
   5.4 Suggestions for future Work .......................... 23

6 Conclusion .............................................. 24

Appendices ............................................... i

A Background ............................................ ii
   A.1 Motion capture in ice hockey .......................... iii
      A.1.1 Concussions in ice hockey ........................ iii
      A.1.2 Motion estimation in 3D ........................... iv
      A.1.3 Multiple camera views ............................ v
      A.1.4 Stereo cameras .................................. v
1 Introduction

Head injuries during high velocity sports are a serious health concern, since they can lead to traumatic brain injuries (TBIs), such as concussions\textsuperscript{1}. The severity of concussions is dependent on the velocity and the direction of impact. Moreover, the risk of TBIs is dependent on the type of sport and many other factors such as the risk of collision in the sport\textsuperscript{1}. Ice hockey has been described as a sport with a high risk for concussions and there is a growing concern and awareness about the injury in the sport\textsuperscript{2}. Ice hockey is the contact sport with the highest incidence rate of concussions\textsuperscript{3}. It has been reported that 2-14\% of all ice hockey injuries and 15-30\% of all ice hockey head injuries are concussions\textsuperscript{4}. In ice hockey, the collisions can be either between two players, between one player and the ice or one player and the rink. With the help of extracting velocities from video analysis, the understanding of the relationship between concussions and the kinematics of the head can be improved by different methods such as for example: finite element simulations, rigid body simulations, instrumented helmets and dummies.

Video analysis to extract 3D velocities from ice hockey has previously been done using multiple camera views\textsuperscript{5, 6}. The drawback with multiple camera views is that several criteria need to be fulfilled: the angles between the camera views should be $90 \pm 30^\circ$, event synchronization needs to be done and the same calibration points need to be visible throughout the video in both angles\textsuperscript{7}. Additionally, some incidents are only filmed from a single camera view. This leads to the problem that many video sequences need to be deselected and therefore many velocities at concussion incidents cannot be investigated\textsuperscript{5}. Therefore it is beneficial if velocities in ice hockey could be extracted from one single camera view.

In this thesis a prototype to extract 3D velocities from a single camera view was developed by using target tracking algorithms and homography. The homography is about finding a transform and project a 2D ice hockey image to an image of the rink model in order to see how the player moves on the ice. Validation of the prototype was done. Additionally, a large number of tackles in ice hockey were investigated where the velocities were extracted for each case. Another method for extracting 3D velocity is the program SkillSpector, which also was validated in this thesis.
2 Aim and objectives

The aim of this project was to develop a prototype to extract 3D velocities from videos in ice hockey by using a single camera angle. This has the higher purpose of gaining a deeper understanding for injuries within the sport and to develop safer protective equipment.

The objectives of this thesis were to:

• Develop a prototype to extract 3D velocities from a single camera angle.

• Calculate how big the error was between the velocities calculated by the prototype compared with an accelerometer.

• Validation of a program called SkillSpector which was used in a previous bachelor thesis. Calculate the error was between the velocities calculated by SkillSpector compared with an accelerometer.
3 Method

A literature study was done in order to make qualitative choices when developing the prototype to extract 3D velocities from the videos with only one camera view. Different target tracking algorithms and point object algorithms in Matlab (MathWorks, Natick, Massachusetts, United States) were studied in order to extract the x and y coordinates in the frames. Pose estimation in DensePose, OpenPose and PoseNet was also investigated with the same purpose of finding the x and y coordinates \[8, 9, 10\]. In order to estimate the z coordinate in the images, different monocular depth maps and homography were examined.

The prototype was implemented and developed using the software Matlab. The full code can be found on GitHub\[3\].

This section begins with the qualitative choices of choosing algorithm for returning object points, choosing tracking algorithm and choosing method for estimating the depth motion (z-coordinate) in the prototype. It also describes the different rink dimensions for the different ice hockey leagues and how the velocity estimation was done. In section 3.5 the prototype and its functions is explained. Further on the validation methods of the prototype and SkillSpector are described. Lastly the criteria for the video collection for the 60 investigated cases are presented.

3.1 Choice of algorithm for insertion of object points

Five different algorithms in Matlab were suitable for returning object points of human origin to follow throughout an ice hockey video: Minimum eigenvalue algorithm, Harris–Stephens algorithm, Features from Accelerated Segment Test (FAST) algorithm and the Binary Robust Invariant Scalable Key point (BRISK) algorithm (described in appendix A.2.4). In order to have these points returned from a desired region, the object region needed to be defined. The desired region could for example be the head or the shoulders in order to track the player. It could also be points on the ice, in order to remove the global motion of the camera from the motion of the player or to create the homography, that will be described later in the report. Another criteria for returning object points, also known as corner

\[1\]https://github.com/beatricebjering/Hockey.git
points, is that they should show a structure that is two dimensional, points of crossings or region overlaps, as shown in the figure 3.1.

![Figure 3.1: Feature points of (a) corners, (b) crossings or (c) overlaps, can be tracked reliably](image)

To evaluate which algorithm that was most suitable for returning corner points from the ice hockey scenes, a qualitative test was done. The *Minimum eigenvalue algorithm* returned many corner points from both the ice hockey players and the ice hockey rink in the frames, more than the Harris-Stephens algorithm. Both the FAST and the BRISK algorithms did not succeed in returning key points from the ice hockey rink. Therefore, the Minimum eigenvalue algorithm was considered in this thesis.

### 3.2 Choice of tracking algorithm

There were two relevant algorithms to follow objects in Matlab, the Kanade-Lucas-Tomasi (KLT) algorithm and the Consciously Adapted Mean Shift (CAMShift) algorithm. In order to follow the objects and the players in the rink by their object points, the KLT algorithm was considered in this thesis. The histogram-based object tracking, that incorporates the CAMShift algorithm, did not follow the objects correctly in the ice hockey videos and was therefore not considered in the prototype. Tracking with the KLT algorithm follows the ice hockey players and the textures in the ice hockey rink accurately since there are many visual textures on the players and on the ice.

The KLT algorithm uses image pyramids that allows the tracker to track the point objects in multiple levels of resolution [11]. If the level of pyramids are increased, the tracker can follow larger displacements of the points between the frames, but at a computational cost. The KLT algorithm is often used for short-term tracking since the points can be lost as the algorithm processes over time, due to light variation, articulated motion or plane rotation [11].
The basic step of the KLT tracking procedure is described in Equation 1. $G$ can be computed from one frame by estimating the image gradients $g$ and their second order moments. $d$ is the displacement of a point $x$ and the solution of the system. $e$ is computed by the difference between two frames $I$ and $J$ and the image gradient. The tracking is defined over a window of pixels, $W$ and $w$ is a weighting function.

$$Gd = e$$

$$G = \int_W gg^Tw \, dA, \quad e = \int_W (I - J)g \, w \, dA$$

### 3.3 Choice of depth estimation

In order to estimate the depth (z coordinate) in the ice hockey videos, homography was used in the prototype. Monocular depth maps were not considered in the prototype since they suffer from too low spatial resolution and could not be used to estimate the depth in the frames. The low spatial resolution is due to the fact that the existing datasets for creating monocular depth maps are trained on situations that are quite different from ice hockey scenarios.

To create an homography of the ice hockey videos, four known point correspondences between the frames and the rink model were needed. The points needed to be visible both in the ice hockey image and on the rink model, on which the ice hockey image was projected. The homography is about finding a transform and project the 2D ice hockey image $I$ to the image of the rink model $J$. The homography translation matrix, $H$ in equation 2, translates the homogeneous points $x_j = (us, vs, s)$ in image $I$ to the points $x = (xs, ys, s)$ in image $J$. $s$ is a non-zero scale factor which is usually set to 1.

$$Hx_j = x$$

The matrix $H$ can be considered as a $3 \times 3$ matrix (equation 3) and is defined up to a scale, depending on the variable $s$. 
\begin{equation}
H = \begin{bmatrix}
h_{11} & h_{12} & h_{13} \\
h_{21} & h_{22} & h_{23} \\
h_{31} & h_{32} & h_{33}
\end{bmatrix}
\end{equation}

Since the matrix $H$ is defined up to scale, the homography has a total number of eight degrees of freedom. Each 2D point in an image is specified in x and y components and therefore the 2D point has two degrees of freedom. This leads to that it is necessary to specify four point correspondences in the images to fully constrain the homography matrix $H$\cite{14}.

### 3.3.1 Rink dimensions

Rink dimensions differ for various hockey organizations, which needs to be taken into account when estimating the velocity on the ice by homography. Below, the standard dimensions for the ice hockey rink in the respective rule book for the Swedish hockey league (SHL) (figure 3.2) and the National Hockey League (NHL) (figure 3.3) are described. The dimensions of the rink features need to be known in order to estimate the velocity of the ice hockey players.
Figure 3.2: Standard rink dimensions in SHL [15].

Figure 3.3: Standard rink dimensions in NHL [16].
3.4 Velocity estimation

Motion estimation between frames can be estimated when an object is moved from \( X = (x_i, y_i, z_i) \) at time \( t_1 \) to \( X' = (x_i + \Delta x, y_i + \Delta y, z_i + \Delta z) \) at time \( t_2 = t_1 + \Delta t \). From this the 3D motion vector, \( \Delta(X, t_1, t_2) = (X' - X) \), can be obtained [17].

In order to estimate the velocities of an ice hockey player in a video sequence with a moving camera, it is essential to distinguish between local and global motion estimation. The global motion are the pixels that are not subject of the local motion. The local motion is the displacement of the objects in the scene, in this case the ice hockey players. By removing the global motion from a scene, the local motion can be obtained [18].

The x and y coordinates were returned by following the ice hockey players in the frames with the object points returned from the Minimum eigenvalue algorithm and tracked by the KLT algorithm. A stationary point close to the player in the scene was also returned and tracked in the same way in order to obtain the global camera motion. The global motion from the camera (\( \Delta x_c, \Delta y_c \)) was subtracted from the motion of the player (\( \Delta x, \Delta y \)) in order to return the local motion of the player (\( \Delta x_l, \Delta y_l \)) in the frames. The z coordinate (\( \Delta z_l \)) was estimated by the homography by taking the difference in length between two points on the ice seen from the camera view. The pixel length was estimated by the known length of the objects on the ice or the ice hockey rink.

The velocity of the ice hockey players was calculated by the local motion (\( \Delta x_l, \Delta y_l, \Delta z_l \)) of the player between each frame and the time between each frame \( \Delta t = t_2 - t_1 \) [19]. This is explained in equation 4.

\[
v = \sqrt{\frac{(\Delta x_l)^2}{\Delta t} + \frac{(\Delta y_l)^2}{\Delta t} + \frac{(\Delta z_l)^2}{\Delta t}} \quad (4)
\]

The impact angle from the 2D vectors (seen from above) was also calculated for each tackle between two players with equation 5.

\[
\cos(\theta) = \frac{(\vec{u} \ast \vec{v})}{(||\vec{u}|| \ast ||\vec{v}||)} \quad (5)
\]
3.5 Prototype

The prototype consists of a main script and several functions and subfunctions, which are described in a flowchart in figure 3.4 below. The functions and subfunctions are described in more detail later.

Main - Reads the video file name and initializes the video readers. Creates a folder with separate images of the video and a graph of the velocities.

defineRegion - Allows the user to define the regions around the attacking player, injured player, 4 points on the ice and a stationary point close to the players. These regions are used to create a number of object points via the Minimum eigenvalue algorithm.

point - Initializes the KLT trackers to track the object points throughout the video. If the validity of the tracked points are not valid, they fall out. The position of the points are saved in matrices for each defined region. In this function, the MaxBidirectionalError can be increased or decreased. The advantage with increasing the MaxBidirectionalError is that there is a bigger chance that the points are tracked in a blurry frame, but to the cost of lower accuracy. If the chosen points are not tracked throughout the sequence the program will not work properly. It is recommended that the MaxBidirectionalError is set to a value between 1-3. The MaxBidirectionalError in the function testpoint2track has to
be the same as in this function.

**point2track** - The user chooses which object point to follow in each region. The correct marker is found with the least square method.

**testpoint2track** - Confirms that the chosen points are being followed correctly by inserting only the chosen points in the video. The *MaxBidirectionalError* can be changed in this function as well. It has to be the same as in the *point* function. The user is asked to insert a point on the ice for the players, in order to correctly project the position on the ice. Since the players can be jumping or moving the body position up or down during the video and to create a correct homography these movements needs to be correlated to the positioning on the ice, which is also taken care of in this function. The projected points on the ice are saved for the two ice hockey players.

**subfunction pixelsize** - This subfunction determines the pixel size of each frame throughout the video. It is used to account for the eventual zoom effect and to estimate the velocity in the function *velocity*. The pixel size for each frame is estimated by defining the pixel length of an object, as close as possible to the players, with a known real-world length in the first and the last frame. The average change is calculated and the average pixel size for each frame is determined.

**Homography** - This function creates the homography of the rink and converts the image coordinates to rink coordinates. The four points chosen on the ice in the function *point*, are used together with four defined points on the rink model to create a geometric transform between the points. The four points on the ice was followed throughout the video sequence with the points returned by the *Minimum eigenvalue algorithm* and tracked by the KLT algorithm. The movement of the two ice hockey players was inserted into the rink model. The center of rotation of the camera was also inserted in order to estimate from which angle the camera is filming the video sequence, in order to correctly estimate the depth of the players movement on the ice.

**subfunction depth** - The width and length of the ice hockey rink is depending on the league. NHL and hockey Canada have the same dimensions, SHL and IIHF (International ice hockey federation) have the same dimensions. The change in depth of the player is estimated by calculating the changes in distance from the camera angle.
velocity - Estimates the velocities of the ice hockey players by using the frame rate of the video, the x and y coordinates of the markers on the players from the frames and the depth from the homography.

3.6 Validation of the prototype

In order to validate the prototype, it was tested on a miniature ice hockey game with miniature ice hockey players. Sensors (LPMS-B2) with accelerometers were attached on the ice hockey players while they were moved along the ice and recording it with a non-stationary mobile phone camera from one angle (figure 3.5). The MaxBidirectionalError in the functions point and testpoint2track was set to 1 in the prototype.

Figure 3.5: Recording with a non-stationary camera

The error between the impact curve of the accelerometer and the curve obtained from the prototype was calculated by taking the difference between each corresponding data point and using the equation 6 below.

\[
error = \frac{\text{accelerometer value} - \text{estimated value}}{\text{accelerometer value}}
\]  

(6)

Further on, the prototype was compared to the velocities estimated with SkillSpector from the previous thesis (seen in table A.1), where a tackle resulted in a concussion. A T-test was done to investigate if there were any significant difference between the methods.
3.7 Validation of SkillSpector

SkillSpector were validated in the same way as the prototype; it was tested on a miniature ice hockey game with miniature ice hockey players and with the sensors (LPMS-B2) attached to the players. SkillSpector reconstructs the 3D movement by using the Direct Linear Transform (DLT). In difference from the prototype validation the incident had to be filmed with two cameras with different angles (90 ± 30 degrees relative to each other can be tolerated) to calculate the velocity in SkillSpector (figure 3.6) \[7\]. The incident was filmed with the same frame rate on both cameras.

![Figure 3.6: Recording with two stationary cameras from different angles (90 ± 30 degrees relative to each other).](image)

The videos was cropped in VirtualDub 1.10.4 (VirtualDub, unknown place of publication) in order to synchronize the event from the two different angles. The videos were analyzed in SkillSpector with the same method as in the previous bachelor thesis \[5\]. Eight calibration points were used. The error between the curve of the accelerometer and the curve obtained from SkillSpector was calculated by using equation 6.
3.8 Video collection

Out of the 60 investigated tackles 40 were collected from YouTube and were found by searching for the players name and "hit". The videos were downloaded with the software YTD Video Downloader 4.0.1 (GreenTree Applications SRL, Bucharest, Romania). The other videos of tackles were provided by SHL. There were also several prerequisites in order to be able to work with the videos:

- The tackle should be visible from at least one camera view.
- The video should not be too blurry. It should be possible to follow the ice hockey players clearly in each frame in order to track them.
- Four known points on the ice hockey rink needed to be visible from one angle throughout the whole video sequence in order to create a homography.

A control of that the injured player suffered from concussion from the tackle was done by a confirmation from trustworthy sources, such as verification on the official websites of SHL or NHL.
4 Results

In the following sections the results are presented. First the velocities from the previous thesis are compared the compiled velocities from the prototype. Secondly the results of the velocities of 60 tackles compiled with the prototype are presented and lastly the validation of the prototype and SkillSpector are explained.

4.1 Velocities in SkillSpector compared with the prototype

Three of the cases from the previous bachelor thesis (video sequences D, G and H, seen in table 4.1) could not be tested in the method because of the bad homography and since the points could not be tracked reliably.

The p-value between the velocities for both the attacking and the injured player in SkillSpector and the prototype was $p=0.07$, showing that there is a significant difference between the two methods. The specific velocities for each case can be found in table 4.1.

Table 4.1: Velocities compiled from the prototype. These are the same cases as for the SkillSpector method, these velocities are written inside the parentheses.

<table>
<thead>
<tr>
<th>Video sequence</th>
<th>Velocity of attacking player [m/s]</th>
<th>Velocity of injured player [m/s]</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>2.80 (8.5)</td>
<td>5.59 (9.1)</td>
</tr>
<tr>
<td>B</td>
<td>9.31 (-)</td>
<td>3.77 (7.7)</td>
</tr>
<tr>
<td>C</td>
<td>4.04 (5.4)</td>
<td>4.99 (3.2)</td>
</tr>
<tr>
<td>D</td>
<td>- (10)</td>
<td>- (3.4)</td>
</tr>
<tr>
<td>E</td>
<td>9,40 (6)</td>
<td>3.67 (5.3)</td>
</tr>
<tr>
<td>F</td>
<td>8.12 (12)</td>
<td>6.42 (20)</td>
</tr>
<tr>
<td>G</td>
<td>- (22)</td>
<td>- (14)</td>
</tr>
<tr>
<td>H</td>
<td>- (-)</td>
<td>- (8.7)</td>
</tr>
<tr>
<td>I</td>
<td>2.20 (7.8)</td>
<td>4.87 (7.2)</td>
</tr>
<tr>
<td>J</td>
<td>5.11 (3.5)</td>
<td>5.11 (2.9)</td>
</tr>
</tbody>
</table>
4.2 Velocities for 60 tackles compiled with the prototype

Velocities from 60 tackles were compiled with the prototype, 30 tackles where concussion occurred and 30 where it did not occur. There was no significant difference in velocities for the attacking players (p=0.89) in the tackles where concussion occurred and did not occur. No significant difference between the velocities of the injured players (p=0.89) for the tackles that resulted in concussion or no concussion was found either.

As seen in table 4.2, the mean velocity in cases where concussion occurred was 6.55 m/s for the attacking player and 4.59 m/s for the injured player. The specific velocities and impact angles for each tackle can be found in table 4.2 and table 4.3 and the velocity distributions for the cases where concussion did and did not occur can be found in figure 4.1 and 4.2. From the tables 4.2 and 4.3 is can also be seen that the mean impact angle in the tackles where a concussion did and did occur were 85.9° versus 75.9°. There were no significant difference (p=0.70) between the impact angles for the two groups.
Figure 4.1: Distribution of velocities for cases where concussion occurred.

Figure 4.2: Distribution of velocities for cases where concussion did not occur.
Table 4.2: Velocities compiled from the prototype where tackles resulted in a concussion.

<table>
<thead>
<tr>
<th>Case</th>
<th>Attacking player [m/s]</th>
<th>Injured player [m/s]</th>
<th>League</th>
<th>Angle °</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-</td>
<td>4.87</td>
<td>NHL</td>
<td>Against plexiglass</td>
</tr>
<tr>
<td>2</td>
<td>8.12</td>
<td>6.42</td>
<td>NHL</td>
<td>47.7</td>
</tr>
<tr>
<td>3</td>
<td>5.75</td>
<td>4.36</td>
<td>NHL</td>
<td>69.3</td>
</tr>
<tr>
<td>4</td>
<td>-</td>
<td>4.99</td>
<td>NHL</td>
<td>Against plexiglass</td>
</tr>
<tr>
<td>5</td>
<td>9.40</td>
<td>3.67</td>
<td>NHL</td>
<td>170</td>
</tr>
<tr>
<td>6</td>
<td>6.90</td>
<td>6.29</td>
<td>NHL</td>
<td>4.60</td>
</tr>
<tr>
<td>7</td>
<td>3.50</td>
<td>1.78</td>
<td>NHL</td>
<td>161</td>
</tr>
<tr>
<td>8</td>
<td>3.65</td>
<td>1.81</td>
<td>NHL</td>
<td>138</td>
</tr>
<tr>
<td>9</td>
<td>-</td>
<td>9.32</td>
<td>IIHF</td>
<td>Against plexiglass</td>
</tr>
<tr>
<td>10</td>
<td>6.97</td>
<td>4.88</td>
<td>SHL</td>
<td>179</td>
</tr>
<tr>
<td>11</td>
<td>9.31</td>
<td>3.77</td>
<td>NHL</td>
<td>96.1</td>
</tr>
<tr>
<td>12</td>
<td>14.3</td>
<td>3.70</td>
<td>SHL</td>
<td>114</td>
</tr>
<tr>
<td>13</td>
<td>6.30</td>
<td>1.75</td>
<td>NHL</td>
<td>5.3</td>
</tr>
<tr>
<td>14</td>
<td>-</td>
<td>9.13</td>
<td>NHL</td>
<td>Against plexiglass</td>
</tr>
<tr>
<td>15</td>
<td>2.80</td>
<td>5.59</td>
<td>NHL</td>
<td>51.8</td>
</tr>
<tr>
<td>16</td>
<td>-</td>
<td>5.11</td>
<td>NHL</td>
<td>Against plexiglass</td>
</tr>
<tr>
<td>17</td>
<td>2.38</td>
<td>1.80</td>
<td>NHL</td>
<td>157</td>
</tr>
<tr>
<td>18</td>
<td>2.47</td>
<td>1.09</td>
<td>NHL</td>
<td>92</td>
</tr>
<tr>
<td>19</td>
<td>-</td>
<td>3.83</td>
<td>NHL</td>
<td>Against plexiglass</td>
</tr>
<tr>
<td>20</td>
<td>3.38</td>
<td>6.43</td>
<td>NHL</td>
<td>73.5</td>
</tr>
<tr>
<td>21</td>
<td>10.3</td>
<td>3.38</td>
<td>SHL</td>
<td>33.9</td>
</tr>
<tr>
<td>22</td>
<td>10.9</td>
<td>9.50</td>
<td>SHL</td>
<td>42.4</td>
</tr>
<tr>
<td>23</td>
<td>7.38</td>
<td>0.81</td>
<td>SHL</td>
<td>135</td>
</tr>
<tr>
<td>24</td>
<td>5.66</td>
<td>2.17</td>
<td>SHL</td>
<td>78.9</td>
</tr>
<tr>
<td>25</td>
<td>3.39</td>
<td>1.68</td>
<td>SHL</td>
<td>23.2</td>
</tr>
<tr>
<td>26</td>
<td>10.5</td>
<td>2.27</td>
<td>NHL</td>
<td>128</td>
</tr>
<tr>
<td>27</td>
<td>6.17</td>
<td>5.79</td>
<td>NHL</td>
<td>179</td>
</tr>
<tr>
<td>28</td>
<td>3.85</td>
<td>2.85</td>
<td>NHL</td>
<td>17.9</td>
</tr>
<tr>
<td>29</td>
<td>5.57</td>
<td>4.08</td>
<td>NHL</td>
<td>62.8</td>
</tr>
<tr>
<td>30</td>
<td>8.22</td>
<td>14.52</td>
<td>NHL</td>
<td>67.3</td>
</tr>
<tr>
<td><strong>Mean value</strong></td>
<td><strong>6.55</strong></td>
<td><strong>4.59</strong></td>
<td></td>
<td><strong>85.9</strong></td>
</tr>
</tbody>
</table>
Table 4.3: Velocities compiled from the prototype where tackles did not result in a concussion.

<table>
<thead>
<tr>
<th>Case</th>
<th>Attacking player [m/s]</th>
<th>Injured player [m/s]</th>
<th>League</th>
<th>Angle °</th>
</tr>
</thead>
<tbody>
<tr>
<td>31</td>
<td>7.08</td>
<td>3.12</td>
<td>Hockey Canada</td>
<td>89.5</td>
</tr>
<tr>
<td>32</td>
<td>5.26</td>
<td>2.44</td>
<td>NHL</td>
<td>165</td>
</tr>
<tr>
<td>33</td>
<td>2.41</td>
<td>0.75</td>
<td>NHL</td>
<td>80.9</td>
</tr>
<tr>
<td>34</td>
<td>3.33</td>
<td>2.94</td>
<td>NHL</td>
<td>20.4</td>
</tr>
<tr>
<td>35</td>
<td>4.51</td>
<td>3.72</td>
<td>NHL</td>
<td>32.8</td>
</tr>
<tr>
<td>36</td>
<td>-</td>
<td>5.92</td>
<td>NHL</td>
<td>137</td>
</tr>
<tr>
<td>37</td>
<td>-</td>
<td>6.18</td>
<td>Against plexiglass</td>
<td>84.9</td>
</tr>
<tr>
<td>38</td>
<td>5.01</td>
<td>5.39</td>
<td>NHL</td>
<td>165</td>
</tr>
<tr>
<td>39</td>
<td>7.64</td>
<td>4.42</td>
<td>NHL</td>
<td>56.9</td>
</tr>
<tr>
<td>40</td>
<td>4.77</td>
<td>3.75</td>
<td>NHL</td>
<td>128</td>
</tr>
<tr>
<td>41</td>
<td>3.07</td>
<td>2.76</td>
<td>NHL</td>
<td>103</td>
</tr>
<tr>
<td>42</td>
<td>3.74</td>
<td>3.83</td>
<td>SHL</td>
<td>82.4</td>
</tr>
<tr>
<td>43</td>
<td>8.80</td>
<td>3.91</td>
<td>SHL</td>
<td>103</td>
</tr>
<tr>
<td>44</td>
<td>7.22</td>
<td>3.07</td>
<td>SHL</td>
<td>80.6</td>
</tr>
<tr>
<td>45</td>
<td>3.66</td>
<td>4.94</td>
<td>SHL</td>
<td>103</td>
</tr>
<tr>
<td>46</td>
<td>3.59</td>
<td>2.57</td>
<td>SHL</td>
<td>59.7</td>
</tr>
<tr>
<td>47</td>
<td>9.83</td>
<td>1.88</td>
<td>SHL</td>
<td>99.2</td>
</tr>
<tr>
<td>48</td>
<td>7.37</td>
<td>7.50</td>
<td>SHL</td>
<td>52.8</td>
</tr>
<tr>
<td>49</td>
<td>8.46</td>
<td>3.75</td>
<td>SHL</td>
<td>51.4</td>
</tr>
<tr>
<td>50</td>
<td>11.5</td>
<td>9.06</td>
<td>SHL</td>
<td>11.1</td>
</tr>
<tr>
<td>51</td>
<td>5.413</td>
<td>3.95</td>
<td>SHL</td>
<td>35.2</td>
</tr>
<tr>
<td>52</td>
<td>8.52</td>
<td>8.23</td>
<td>SHL</td>
<td>50.1</td>
</tr>
<tr>
<td>53</td>
<td>11.3</td>
<td>7.19</td>
<td>SHL</td>
<td>16.4</td>
</tr>
<tr>
<td>54</td>
<td>3.99</td>
<td>3.66</td>
<td>SHL</td>
<td>101</td>
</tr>
<tr>
<td>55</td>
<td>11.0</td>
<td>7.75</td>
<td>SHL</td>
<td>26.6</td>
</tr>
<tr>
<td>56</td>
<td>4.58</td>
<td>1.80</td>
<td>SHL</td>
<td>108</td>
</tr>
<tr>
<td>57</td>
<td>7.55</td>
<td>7.00</td>
<td>SHL</td>
<td>94</td>
</tr>
<tr>
<td>58</td>
<td>13.5</td>
<td>2.14</td>
<td>SHL</td>
<td>179</td>
</tr>
<tr>
<td>59</td>
<td>3.80</td>
<td>5.03</td>
<td>SHL</td>
<td>67.2</td>
</tr>
<tr>
<td>60</td>
<td>3.68</td>
<td>6.47</td>
<td>SHL</td>
<td>143</td>
</tr>
<tr>
<td><strong>Mean value</strong></td>
<td><strong>6.45</strong></td>
<td><strong>4.44</strong></td>
<td></td>
<td><strong>75.9</strong></td>
</tr>
</tbody>
</table>
4.3 Validation of the prototype

The prototype was validated with a miniature hockey players moving with velocities of 0-2.3 m/s. The average error for each test for the data points between the accelerometer and the estimated values through the prototype is described in Table 4.4. Out of the 11 tests that were made, the mean error was estimated to 21.7%. The errors were in the range of 14.6-35.1%.

Table 4.4: Average error for each test between the velocities estimated by the prototype and the accelerometer.

<table>
<thead>
<tr>
<th>Test nr</th>
<th>Average error (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1a</td>
<td>25.9</td>
</tr>
<tr>
<td>2a</td>
<td>35.1</td>
</tr>
<tr>
<td>3a</td>
<td>24.0</td>
</tr>
<tr>
<td>4a</td>
<td>14.6</td>
</tr>
<tr>
<td>5a</td>
<td>25.6</td>
</tr>
<tr>
<td>6a</td>
<td>23.6</td>
</tr>
<tr>
<td>7a</td>
<td>14.7</td>
</tr>
<tr>
<td>8a</td>
<td>27.3</td>
</tr>
<tr>
<td>9a</td>
<td>14.5</td>
</tr>
<tr>
<td>10a</td>
<td>21.9</td>
</tr>
<tr>
<td>Mean error</td>
<td>21.7</td>
</tr>
</tbody>
</table>
4.4 Validation of SkillSpector

SkillSpector was validated with a miniature ice hockey players moving with velocities of 0-2 m/s. The average error, between each test, for the data points between the accelerometer and the estimated value through SkillSpector is described in table 4.5. Out of the six test that were made the mean error was 37.4%, in the range of 19.5-47.3%.

Table 4.5: Average error for each test between the velocities estimated by the SkillSpector and the accelerometer.

<table>
<thead>
<tr>
<th>Test nr</th>
<th>Average error (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1b</td>
<td>43.8</td>
</tr>
<tr>
<td>2b</td>
<td>19.5</td>
</tr>
<tr>
<td>3b</td>
<td>44.5</td>
</tr>
<tr>
<td>4b</td>
<td>47.3</td>
</tr>
<tr>
<td>5b</td>
<td>40.9</td>
</tr>
<tr>
<td>6b</td>
<td>28.1</td>
</tr>
<tr>
<td>Mean error</td>
<td>37.4</td>
</tr>
</tbody>
</table>
5 Discussion

The developed prototype can be a tool to extract velocities in ice hockey in order to gain a better knowledge of the relation between the kinematics of the head and concussions. This might help in the development of new rules or protective equipment to serve the higher purpose of a better health for the ice hockey players. The prototype is developed for shorter videos.

In this section the results from the prototype and SkillSpector will be discussed.

5.1 Performance and sources of error of the prototype

There are several advantages with the developed prototype in this thesis. The velocities can be estimated from a single camera view, which is a benefit since many incidents only are filmed from one camera view and therefore the velocities cannot be estimated by other methods such as the DLT algorithm (used in SkillSpector) and stereo cameras. This leads to that many more incidents can be investigated. Another advantage with the prototype is that large parts of the program is automatic, the tracking of the object points and points on the ice are being tracked automatically throughout the sequence after these points have been chosen manually. Though there is still room for improvements.

The resulting velocities depends on how the user inserts the points on the ice and how precise the homography gets. It also depends on which points the user chooses to track on the ice hockey players. Depending on how large area the user defines in the function defineRegion and how many corner points that are detected, the method will take longer or shorter time to execute. The more corner points that are found, the longer time it will take. Another disadvantage is that the players can not be followed if they are covered by something in a frame, such as another player or by a post. It can also be difficult to track a point if the frames are too blurry, since the corner points (figure 3.1) can not be identified by the algorithms. This can partially be adjusted by the MaxBidirectionalError in the functions point and testpoint2track where the error can be increased and decreased. If the error is increased, it is a bigger chance that the points are tracked in a blurry frame, but to the cost of lower accuracy.
The performance validation shows that the prototype has a mean error of 21.7%, with a range of 14.6-35.1% (table 4.4). This range might depend on how good the homography was and how precise the points on the players and the ice was being followed. An additional human error can be how the user denoted the length of the known length object in the function pixellength that returns the pixel length in the frames.

### 5.2 Evaluation of SkillSpector

It could be seen that the velocities compiled with the prototype and SkillSpector (table 4.1) had a significant difference. It is difficult to say what the difference depends on since there are multiple error sources. One of the error sources could be in the prototype, since the mean error is 21.7% (table 4.4). It could also be an error in the use of SkillSpector or in the program itself, where the mean error was 37.4% (table 4.5). The higher mean error in SkillSpector might be due to that this method is more dependant on how the user inserts the points to follow the players, since these point needs to be inserted on the player in each frame. The insertion of the eight calibration points in the videos might also be a source of error. Additionally, another source of error might in the event synchronization that had to be done for the two different camera angles. SkillSpector is less automatic than the prototype developed in this thesis and therefore it takes longer time to process the incidences in SkillSpector, which is another disadvantage.

### 5.3 Velocities estimated by the prototype

Out of the 60 velocities estimated from the prototype no significant difference was found between the attacking nor the the injured player between the cases where concussion did and did not occur (from tables 4.2 and 4.3). Additionally no significant difference between the impact angles for the two groups could be found. This could be an indication that the velocity is not the main factor that is crucial if a concussion will occur in a tackle or not. Other factors such as rotation and which body parts that are involved in the incident might have a higher impact on the occurrence of concussion. There could also be a risk that the league has chosen to not announce that the player suffered from concussion, due to privacy
reasons. This could be a source of error to the 30 tackles in Table 4.2 and might result in hidden statistics of the number of concussions.

5.4 Suggestions for future Work

There are definitely room for improvements in order to minimize the error in the prototype. Many of the sources of error in the prototype depend on the human error. It would therefore be beneficial to minimize the human input, especially in the estimation of the pixel length, zoom effects and homography. It would also be interesting to see if a better result would be achieved if another method was used to estimate the depth motion of the player. For this thesis homography is used for this purpose, but if depth maps in the ice hockey environment could give an accurate estimation in the future this would also be relevant. A way to track the players even if they were hidden behind another object could also be something to improve in the current prototype. Another interesting thing would be to evaluate the method on higher velocities, perhaps on a real ice hockey rink since this was not possible to archive in the validation model for this thesis.

The basics of the developed prototype could also be applied to estimate velocities in other sports with known length objects on the ground, which might be an application in future studies.
6 Conclusion

A prototype to extract 3D velocities from a single camera view from videos in ice hockey was developed. The results shows that the developed prototype had an average accuracy of approximately 21.7% when comparing the values with an accelerometer. A quantitative evaluation of 60 ice hockey tackles was made (30 with confirmed concussion, 30 without concussion). There were no significant differences in velocities for either the attacking players or the injured players in the cases where concussion did or did not occur.

Not all video sequences from the previous bachelor thesis could be processed with the prototype due to some constraints with the tracking of the players. It can be concluded that there is a significant difference between the velocities compiled with SkillSpector and the prototype in this thesis as the validation of SkillSpector show that it had a mean error of 37.4%.
Appendices

A  Background

Introduction

The background will first begin with an introduction of the subject *concussions in ice hockey*, explaining the mechanism of injury. The first section is also explaining how 3D velocities previously have been estimated by techniques using multiple camera views. From section A2, the report will focus on how to find the 3D velocities of the ice hockey players from only one camera view. Section A2 describes how to find the x and y coordinates in the images and section A3 describes how to find the z-coordinate, the depth, in the images.
A BACKGROUND

A.1 Motion capture in ice hockey

Head injuries during high velocity sports is a serious health concern, since it can lead to TBI’s, such as concussions [1]. The severity of concussions depends on, for example, the velocity and the direction of impact. With the help of extracting velocities from video analysis, the understanding of the relation between concussions and the kinematics of the head can be improved by, for example: finite element simulations, rigid body simulations, instrumented helmets and dummies.

This section is first describing the mechanism of injury and the background of concussions in ice hockey. Secondly, motion estimation in 3D will be described shortly. Further on, motion estimation from multiple camera views will be explained. Lastly, velocities from a previous study where the players suffered from concussion will be presented.

A.1.1 Concussions in ice hockey

The risk of TBI is dependant on the type of sport and many other factors. Most importantly, the risk of injury is increased in high-velocity sports where there is a risk of collision [1]. The collisions can be either between two players, between one player and the ice or one player and the rink. Ice hockey has been described as a sport with high risk for concussions and there is a growing concern and awareness about the injury in the sport [2]. Ice hockey is the contact sport with the highest incidence rate of concussions [3]. It has been reported that 2-14 % of all hockey injuries and 15-30 % of all hockey head injuries are concussions [4].

Concussion is a diffuse injury that occurs in a widespread area over the brain. There are several different definitions for what a concussion is. One definition from the 5th International Conference of Concussions in Sport is “Concussion is a traumatic brain injury induced by bio-mechanical forces...”[20]. Additionally, an impulsive force transmitted to the head from the face, head, neck or other parts of the body, induces concussions. The brain deforms quite easily when shear forces are applied, which can be generated when acceleration or deceleration forces are applied to the skull [4]. The inertial forces affecting the brain can be either linear or angular acceleration (or deceleration) [21]. Linear acceleration
is a movement in a straight line that passes through the head’s centre of gravity. Angular acceleration, in comparison, is when the head is accelerated tangentially. It has been shown that the latter type of acceleration is solely responsible for the loss of consciousness in a concussion [21].

The brain is typically shielded from colliding with the walls of the skull by the cerebrospinal fluid and external membranes. Though a forceful impact on the brain can result in that the brain comes in contact with the skull. This can cause deformation, distortion or compression of neural tissue, which initiates a cascade of molecular events that disrupts the normal cell function in the brain [21], [22].

If a player has a history of concussions the risk for future concussions is increased [23]. Usually the injured player recover within one month but concussions can also lead to long term disability. The understanding of concussions has improved in recent years with different methods such as video analysis, rigid body simulations and instrumented helmets [7].

A.1.2 Motion estimation in 3D

Motion estimation in the video context can generally be described as the changes in pixel value in an position between two or more frames. By using different methods, velocities can be withheld from prerecorded videos in both 2D and 3D format. 2D velocities can be calculated from only one camera view by estimating the changes in the x and y coordinates in the frames. This method makes it possible to capture the 2D velocity of the ice hockey players quite easily, though the 2D velocity does not represent the true 3D movement of the player.

3D motion analysis shows the true 3D movements of the body, but can be a bit more tricky to estimate than the 2D motions. In order to estimate the 3D movements the x, y and z coordinates of the target needs to be estimated. This can be done in different ways, such as for instance stereo cameras, motion tracking systems, camera calibration and various video analysis methods.
A.1.3 Multiple camera views

One method to extract 3D velocities is to use multiple camera views to find the x, y and z coordinates of the target. The optimal angle between the cameras is 90 degrees, though a deviation of ± 30 degrees from this angle can be tolerated. Additionally, event synchronization needs to be done to link the event that is recorded in the cameras.

To reconstruct the 3D movement, several algorithms can be used. One of the most common algorithm is the DLT algorithm, which is common when the camera positions are flexible. This is mostly the case in sports. At least six calibration points for each camera view are needed to construct the DLT parameters. These calibration points needs to be visible from each camera view. Once these parameters are constructed, the movement of the chosen points on the body can be reconstructed and the velocities of these points can be calculated. The reconstructed movement points is only guaranteed to be accurate within the calibration region. SkillSpector (Video4Coach., Odense, Danmark) is an example of a tool that can and has been used to analyze 3D movements from multiple camera views in ice hockey.

A.1.4 Stereo cameras

Stereo cameras is a method that has been extensively studied in computer vision. It is a camera with multiple separate camera lenses and image sensors that can be used to extract the 3D coordinates of an object in an image. There are two different cases of calculating the depth with stereo cameras; parallel stereo cameras and convergent stereo cameras. To calculate the disparity and the depth in the image with parallel cameras, the cameras needs to have parallel optical axes, a known baseline as well as a known focal length. In the case of convergent stereo cameras, the cameras needs to be calibrated and the camera matrices needs to be known.

The drawback with this method is that you need to record the video sequences with stereo cameras from start. Moreover, the drawback with using multiple camera views is that all the specific criteria that previously has been described, needs to be fulfilled. This leads to that many video sequences needs to be deselected and
therefore many velocities at concussion incidents cannot be investigated [5]. As a consequence of this, it would be beneficial if the velocities in ice hockey could be extracted from one single camera view. From chapter 2 and onward, finding the velocity of the ice hockey players from one camera view will therefore be considered.

A.1.5 Previous studies on velocities in ice hockey

A previous study, with the method of using multiple camera views to calculate velocities in SkillSpector, shows that the velocities in a tackle where one of the players suffered from concussion is significantly higher than in a tackle that didn’t result in any concussion [5]. Additionally it also shows that the most commonly involved body parts in a tackle that resulted in a concussion was: shoulder, yaw and head. The velocities of the attacking and the injured player estimated from this method are shown in Table A.1 below.
Table A.1: velocities compiled from SkillSpector, from previous bachelor thesis, for both the injured and attacking players.

<table>
<thead>
<tr>
<th>Video sequence nr</th>
<th>velocity of attacking player [m/s]</th>
<th>velocity of injured player [m/s]</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>8.5</td>
<td>9.1</td>
</tr>
<tr>
<td>B</td>
<td>-</td>
<td>7.7</td>
</tr>
<tr>
<td>C</td>
<td>5.4</td>
<td>3.2</td>
</tr>
<tr>
<td>D</td>
<td>10</td>
<td>3.4</td>
</tr>
<tr>
<td>E</td>
<td>6.0</td>
<td>5.3</td>
</tr>
<tr>
<td>F</td>
<td>12</td>
<td>20</td>
</tr>
<tr>
<td>G</td>
<td>22</td>
<td>14</td>
</tr>
<tr>
<td>H</td>
<td>-</td>
<td>8.7</td>
</tr>
<tr>
<td>I</td>
<td>7.8</td>
<td>7.2</td>
</tr>
<tr>
<td>J</td>
<td>3.5</td>
<td>2.9</td>
</tr>
</tbody>
</table>

A.2 Target tracking

Target tracking is a broad field in computer vision and has been used in several sports to track either the players or the ball in sports such as football, basket and volleyball [25], [26], [27], [28], [29]. This section will focus on finding the x and y coordinates of the ice hockey players from one single camera view. First basic knowledge in motion estimation of ice hockey videos will be discussed, and then different methods to track targets by key points from automatic pose estimation and different tracking algorithms in Matlab will be examined. Lastly, point object algorithms that are needed in order to return point objects from the images are described. Finding the z-coordinate will be discussed in chapter A3.

A.2.1 Local and global motion estimation

Some particular challenges in ice hockey videos are that a single camera only shows a part of the rink and the players, as well as that the camera moves. In order to estimate the velocities of an ice hockey player in a video sequence with a moving camera, it is essential to distinguish between local and global motion estimation. The global motion are the pixels that are not subject of the local motion. The local motion is the displacement of the objects in the scene, in this case the ice hockey players. By removing the global motion from a scene, the precise local motion can be obtained [18]. The global motion in the videos investigated in this thesis, comes
from the moving camera. The considered camera-based motions in this thesis is panning and zooming.

Motion estimation between frames can be estimated when an object is moved from $X = (x_i, y_i, z_i)$ at time $t_1$ to $X' = (x_i + \Delta x, y_i + \Delta y, z_i + \Delta z)$ at time $t_2 = t_1 + \Delta t$. From this the 3D motion vector, $\Delta(X, t_1, t_2) = (X' - X)$, can be obtained \footnote{17}.

The magnitude of the velocity of the ice hockey players was calculated by the local motion $(\Delta x_l, \Delta y_l, \Delta z_l)$ of the player between each frame the time between each frame $\Delta t = t_2 - t_1$ \footnote{19}. This is seen in equation 7.

$$v = \sqrt{\frac{(\Delta x_l)^2}{\Delta t} + \frac{(\Delta y_l)^2}{\Delta t} + \frac{(\Delta z_l)^2}{\Delta t}}$$  \hspace{1cm} (7)

\section*{A.2.2 Pose estimation}

Single or multiple human pose estimation from a monocular camera is a relatively new recent field of work in computer vision. The human figures and key body joints can be estimated from images and videos through neural networks \cite{30}, \cite{8}. These key points can for example include the head, shoulders, elbows, eyes, hips, hands and feet. A neural network is a type of machine learning that allows the computer to learn via an algorithm and incorporating new data \cite{31}. They are mostly trained with data sets consisting of different pictures of a certain type. By analysing the reoccurring patterns in these images the network learns to categorise new images.

DensePose is an open source network that maps pixels from a single or multiple human bodies in a 2D RGB image, and transforms it into a 3D based model \cite{8}. The extraction of the key points and the 3D estimation of the original image (figure A1) from DensePose can be seen in figure A2 and figure A3. Researchers have mapped human body parts through a neural network by studying 50,000 Common objects in Context (COCO) images. From this system the x and y coordinates from the key points in the image can be obtained. The 3D points on the surface model is estimated by the network.

OpenPose is another open source library for real-time multi-human pose detection in 2D \cite{9}. Another real-time pose detection is TensorFlow’s version of
PoseNet [10]. It can also detect multiple or single persons in an image and extract the 2D positions of the keypoints in the image.

### A.2.3 Target tracking algorithms

Matlab is an efficient tool with several algorithms for object tracking. Two of the relevant algorithms from the computer vision system toolbox for this project are:

**Histogram-based object tracking** that incorporates the CAMShift algorithm for tracking the object and histogram pixel values to identify the tracked object. CAMShift saves the colour histogram of the object in the initialised search window as a reference to find the object in the next frames. The search window can also be adjusted adaptively, which is an advantage since it can handle dynamically changing colour distribution [32].

**The KLT algorithm** is another way to track objects in a scene by tracking point objects. The algorithm works particularly well for objects that have a visual texture and that doesn’t change shape [32]. The KLT algorithm uses image pyramids that allows the tracker to track the point objects in multiple levels of resolution [32]. If the level of pyramids are increased, the tracker can follow larger displacements of the points between the frames, but at a computational cost. In order to track point objects the corner points from the images needs to be returned, which will be described in the next section.
A.2.4 Point object algorithms

There are several algorithms that can detect image features and return point objects that store pixel information. The point objects determine how well the motion in the frames can be estimated. The points can be lost along the frames, due to articulated motion, variation in light and plane rotation. The neighborhood of the feature should show a structure that is two dimensional, which is the case of corner points (figure A4a) [33]. Points of crossings (figure A4b) or where several regions overlap (figure A4c) can also be tracked reliably.

Figure A.4: Feature points of corners (a), crossings (b) or overlaps (c), can be tracked reliably

For point tracking with the KLT algorithm, the following algorithms in Matlab is suitable to return point objects from scenes of human origin [34]:

- **Minimum eigenvalue algorithm** - Returns corner points from a grey-scale image. It can handle image registration with little or no scale change.

- **Harris–Stephens algorithm** - Returns corner points and is more efficient than the minimum eigenvalue algorithm. It can handle image registration with little or no scale change.

- **FAST algorithm** - Return corner points from a grey-scale image. Can handle image registration with little or no scale change.

- **BRISK algorithm** - Detects multiscale corner features and return BRISK points objects from a grey-scale image. Apart from the other algorithms it can handle changes in scale and rotation.
A.3 Methods for depth estimation

This section will focus on how the depth (z-coordinate) of the ice hockey players from one single camera view can be estimated. The depth parameter is essential to calculate the 3D velocity. Various methods to achieve this will be presented below.

A.3.1 Monocular depth maps

Many existing approaches of estimating depths in images relies on an approach that the scene of interest is available in multiple views points, stereo cameras, different lighting conditions, assumptions of a fixed camera or a static scene [30]. In the case of having only one single input image, monocular depth estimation can be an alternative to estimate the changes in the z-coordinate of a player in an image. Estimating depth from a single image requires monocular depth cues such as object sizes, image position, visual angles and perspective [35]. Convolutional neural networks (CNNs) has brought advancements in the field of single depth estimation [36], [37], [38], [30]. These neural networks are trained on large amounts of data. Most of these networks are trained on standard datasets such as KITTI, NYU or Make3D [39]. Moreover, these datasets are usually collected in specific scenarios and captured by using laser scanners or RGB-D sensors, such as Kinect. The NYU dataset is comprised of various indoor scenes, KITTI is comprised by various city scenes with both static and dynamic objects and Make3D is comprised a small number of training sets on both outdoor and indoor scenes [40], [41], [42], [39]. Another important thing to mention is that many of the existing monocular depth maps methods suffer from low spatial resolution [36]. Additionally the datasets are trained on situations that are quite different from ice hockey scenarios.

Monodepth is an unsupervised deep neural network developed by Facebook that can predict pixel wise depth for a single image despite the absence of ground truth data [30]. An example of a depth map from Monodepth can be seen in figure A5. While training the network stereo images are needed. The main idea behind this method is to learn to produce the right image from the left to find the 3D shape of the scene. The method is fast and can create the dense map in 35 milliseconds for a 512 x 256 image.
Another unsupervised network, that is developed by Google, is able to predict depth from a scene as well as the "ego motion" (the motion of the camera between two frames) \cite{43}. The object and camera motion are learned from monocular videos as input. The method improves the depth estimation in scenes with a lot of motion, compared to other state-of-the-art approaches \cite{43}.

Semi-supervised networks are also a state-of-the-art approach that make use of both supervised and unsupervised training cues \cite{44}. This method benefits from the ground truth measurements cues from the supervised training cues for the actual depth in a scene. A huge amount of training data can be obtained by the unsupervised training cues, since this data is much easier to obtain. The network is able to predict depth maps on thin and distant objects.

### A.3.2 Homography

Homography, also called projective transformation often occurs when working with images. It is a method that transforms points from one particular space to another space, given that the spaces shares the same image center or depict planar surfaces \cite{14}. If it is known that all world points are in one plane, it is possible to determine how an image coordinate is going to be mapped in the real world coordinates. Moreover, it does not require calibration between the two cameras that depicts the different scenes.

Homography plays an important role in camera calibration where the world is assumed to be a planar surface ($Z = 0$) and where a pinhole camera model is used \cite{45}. The pinhole camera is the most common and simplest camera model. In this model the light from an object enters through a small pinhole and produces a image on an image plane that is set behind the pinhole (see figure A6). By using homogeneous coordinates, the relationship between the image coordinates and
the world coordinates can be specified by the homography and the point on the image plane can be transformed to a point in the world coordinate system.

![Figure A.6: The pinhole camera model](image)

Homography can also find a way to transform a set of points from an image $I$ to another image $R$. Similar methods have been used in other sports to project an image to the various play field models, both in ice hockey and in other sports such as basketball and football [25], [27], [46], [47], [48]. In this thesis, the homography is about finding a transform and projecting the 2D ice hockey image $I$ to the image of the rink model $J$. By this method it is possible to estimate how much the player moves in the $z$-coordinates. The homography translation matrix, $H$ in equation 8, translates the homogeneous points $x_j = (u_s, v_s, s)$ in image $I$ to the points $x = (x_s, y_s, s)$ in image $J$. $s$ is a non-zero scale factor which is usually set to 1 [13].

$$Hx_j = x \quad (8)$$

The matrix $H$ can be considered as a $3 \times 3$ matrix (equation 9) and is defined up to a scale, depending on the variable $s$.

$$H = \begin{bmatrix}
h_{11} & h_{12} & h_{13} \\
h_{21} & h_{22} & h_{23} \\
h_{31} & h_{32} & h_{33}
\end{bmatrix} \quad (9)$$

Since the matrix $H$ is defined up to scale, the homography has a total number of eight degrees of freedom. Each 2D point in an image is specified in $x$ and $y$ components and therefore the 2D point has two degrees of freedom. This leads to that it is necessary to specify four point correspondences in the images to fully constrain the homography matrix $H$ [14]. From the visible lines and points on the
ice, correspondences between the geometric rink model and the ice hockey images can be found, and the homography can be estimated. Below an example can be seen on how a set of points on the ice have been used to transform and project figure A7 on figure A8 by homography.

If more than four point correspondences are used to find the homography matrix $H$, several solutions exist. Therefore it is necessary to find a method that determines the best solution. One solution to this is using some sort of cost function. Though, this approach have situations where false point correspondences occur and this will interfere with the computation of the homography matrix $[13, 49]$. These false matches are called outliers and needs to be eliminated so that the homography is estimated only by using inlier matches. The outliers can be eliminated by the following methods:

**RANSAC**, Random Sample Consensus, is a method where a model is scored by a number of data points that are set within a threshold distance. It builds on the idea that for a number of iterations, a random sample of four point correspondences are selected and from which the homography matrix are computed. The correspondences are classified as either inliers or outliers, and after all the iterations are done the iteration containing most inliers is chosen. The homography is then recomputed based on the inliers in that iteration $[49]$. In this model an error threshold $t$ needs to be set in order to determine the probability that the point is and inlier. Additionally the number of iterations $N$ needs to be determined to ensure that at least one of the random samples are free from outliers. $N$ can be chosen by $N = \frac{\log(1 - p)}{\log(1 - (1 - \epsilon)^s)}$ $[49]$. The variable $s$ is the correspondences in each iteration, which in this case will be four. $\epsilon$ is the probability that the sample correspondence is an outlier and this variable can be adaptively determined together with $N$.

**LMS**, Least Median of Squares, is based on the median distances to all the points
in the dataset to score the model [13]. The model with the least median is then selected. The advantage of the LMS is that no prior knowledge of the error is required. Moreover it works very well if there are less then 50% outliers. Though if more than 50% of the data set consists of outliers, the median distance would be to an outlier correspondence instead of the inlier correspondence.
References


REFERENCES


