[Resources to help use this Word template to create your thesis can be found at:

[Training in using Word to create a thesis](https://www.southampton.ac.uk/doctoral-college/professional-development-programme/about-the-programme/knowledge-and-techniques-for-research/using-it-software/text-word.page)

Web pages

<http://go.soton.ac.uk/thesispc>

or

<http://go.soton.ac.uk/thesismac>]

**UNIVERSITY OF SOUTHAMPTON**

FACULTY OF [YOUR\_FACULTY (in capitals)]

[Academic\_Unit (underlined)]

Volume [X] of [Y]

[Thesis\_Title (bold)]

by

**[Your\_Name (bold)]**

Thesis for the degree of [name of degree]

[Month\_Year]

**UNIVERSITY OF SOUTHAMPTON**

**ABSTRACT**

FACULTY OF [YOUR\_FACULTY (in capitals)]

[Discipline (underlined)]

Thesis for the degree of [Doctor of Philosophy\_or\_something]

**[THESIS\_TITLE (bold and in capitals)]**

[Your\_Full\_Name e.g. Arthur Francis Jones]

There are many different types of motion and differentiating them can help us better understand the scene we observe. This report described the first attempt at disambiguating different motions through acceleration detection in computer images. We decompose acceleration into resultant, tangential and radial acceleration by geometry. The information of radial acceleration on leading foot can be used to detect Heel-Strike in gait and it performs excellent in the experiments.

Table of Contents

[Table of Contents i](#_Toc498004668)

[Table of Tables i](#_Toc498004669)

[Table of Figures i](#_Toc498004670)

[List of Accompanying Materials i](#_Toc498004671)

[Academic Thesis: Declaration Of Authorship i](#_Toc498004672)

[Acknowledgements i](#_Toc498004673)

[Definitions and Abbreviations i](#_Toc498004674)

[Chapter 1 Use the style Heading 1 for chapter titles. There is numbering attached so only use it for chapter titles 1](#_Toc498004675)

[1.1 Use the style Heading 2 for subheadings. This style has numbering attached 1](#_Toc498004676)

[1.1.1 Use the style Heading 3 for the next level of headings. This style has numbering attached. 1](#_Toc498004677)

[1.2 Images 1](#_Toc498004678)

[1.3 Tables 1](#_Toc498004679)

[Chapter 2 Working efficiently 1](#_Toc498004680)

[2.1 Learn useful ways to work with your file 1](#_Toc498004681)

[Chapter 3 A new chapter 1](#_Toc498004682)

[3.1 Adding new sections 1](#_Toc498004683)

[3.2 Cross-referencing 1](#_Toc498004684)

[Chapter 4 Reviewing tools 1](#_Toc498004685)

[4.1 Track changes 1](#_Toc498004686)

[4.2 Spell checker and auto-correct 1](#_Toc498004687)

[Chapter 5 Chapter 5 awaits content 1](#_Toc498004688)

[5.1 Chapter 5 subheading 1](#_Toc498004689)

[Chapter 6 Finishing 1](#_Toc498004690)

[6.1 Printing and PDFs 1](#_Toc498004691)

[Appendix A Your first appendix 1](#_Toc498004692)

[Appendix B Your second appendix 1](#_Toc498004693)

[Glossary of Terms 1](#_Toc498004694)

[List of References 1](#_Toc498004695)

[Bibliography 1](#_Toc498004696)

Table of Tables

**No table of figures entries found.**

Table of Figures

[Figure 1 The Clipboard group in Word 2010 1](#_Toc498004497)

List of Accompanying Materials

Academic Thesis: Declaration Of Authorship

I, [please print name]

declare that this thesis and the work presented in it are my own and has been generated by me as the result of my own original research.

[title of thesis]

I confirm that:

1. This work was done wholly or mainly while in candidature for a research degree at this University;
2. Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated;
3. Where I have consulted the published work of others, this is always clearly attributed;
4. Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work;
5. I have acknowledged all main sources of help;
6. Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself;
7. [Delete as appropriate] None of this work has been published before submission [or] Parts of this work have been published as: [please list references below]:

Signed:

Date:

Acknowledgements

Definitions and Abbreviations

“One picture is worth more than ten thousand words”

Anonymous

Then what about two pictures, three pictures…

# Introduction

## Context

It not been isolated with any success before.

An image is a snapshot in which all motions are frozen in time. This implies that video involves many motions which coalesce to form the image sequence. Nowadays, computer vision approaches can differentiate objects in motion from those which are static, but little more [1], [2]. In actuality there are many different types of motion: in the simplest sense there are objects that move with velocity and some that move with acceleration; many objects have more complicated motions. This paper describes a new approach designed to disambiguate different types of motion starting with the detection of objects moving with acceleration via established computer vision techniques.

|  |  |
| --- | --- |
| 13763-walking_news.jpg | ../Screen%20Shot%202016-06-21%20at%2008.41.13.png |
| Figure 1.1 Different types of motion[[1]](#footnote-1) | |

Figure 1.1 illustrates the diversity of motion, the man in the left image is walking with constant velocity in general and the athlete on the right is speeding up, or accelerating. Each part of both men is experiencing different types of motion, especially the legs. Imagine a person is walking, the body is approximately at a constant velocity, and one of the legs is stationary to support the body while the other one is swinging forward like a pendulum, as shown in Fig. 2. These motions can be identified by acceleration because once the status of an object has changed, there must be acceleration. Therefore, we hypothesise that we can find the legs of a person’s body and discriminate between them by extracting their acceleration features.

|  |
| --- |
| ../Screen%20Shot%202016-06-22%20at%2021.57.39.png |
| Figure 1.2 A walking cycle [3] |

Apart from identifying behaviours, acceleration also offers an alternative approach to understand image content. For example in autonomous driving, the acceleration of other cars can help a computer to make correct decisions about other vehicle’s motion trajectories.

## Contributions

* We extend the original Horn Schunck method optical flow technique to focus on acceleration.
* Our analysis shows the constraints to be too stringent for application in real-world video footage, so we then explore the use of the other state of art optical flow algorithms as basis approximating acceleration with wider applicability in general video.
* We show generalised application of acceleration detection in a richer selection of imagery for not only the new approach but also the new basis, on synthetic and on real image sequences.
* We compared the capability of discriminating motion characteristics between velocity and acceleration flow and show the advantages of acceleration.
* We evaluate the sensitivity of our approach to different imaging conditions via a wider range of datasets;
* We now evaluate different types of distortion: visual angle, lighting condition, Gaussian noise, occlusion and low resolution; and
* We compare the performance of our new operator with that of a previous technique and show performance improvement and capabilities.

## Publications

## Thesis Outline

Table of content

# Databases

## The Difficulties of Acceleration Databases

New techniques benefited greatly by the emerge of challenging databases. These databases push the limits of the present techniques and expose where they fail, stir the inspiration of new techniques. Therefore well designed databases are essential for quantitatively evaluation of the new algorithm. For acceleration, it is difficult to find a proper database since previous research did not notice the higher order of motion flow contain different information. As far as we know, there are no databases designed for acceleration or even higher order motion flow algorithm evaluation. Therefore, we build a new database by our own for evaluating acceleration algorithm in this thesis. Before illustrating the results, it is worth to describe how these images datasets are created.

## Synthetic Images Database for Acceleration

The advantage of synthetic images is that the input signal is clean: without motion blur, specularity, rigid motion or other types of noise. Also, the motion field and scene properties can be manipulated as required. This enable us to test whether we can detect acceleration and ground truth can be obtained easily.

Firstly, we use five shape masks with different motion in a sequence of synthesised images as shown in Figure 2.1(a). These contain: one stationary shape; an upper circle which is moving with constant velocity along the x axis; the lower circle is also moving horizontally but with constant acceleration; the smaller triangle is rotating with constant angular velocity (rotation incorporates radial acceleration); and the larger triangle is a pendulum which undergoes different types of motion during a period (when the massive bob at equilibrium position, the pendulum has maximum velocity and zero acceleration; when it swinging to the sideways, the accelerating is increasing while velocity is decreasing until ceases to zero). Texture is essential for optical flow to detect motion so these objects and background are all randomly textured and the texture is fixed during the image sequence, as shown in Figure 2.1(b). The objects are not visible in the textured image but if we view the video composed of these synthesised images, we can perceive the objects as moving.

|  |  |
| --- | --- |
| ../Python/pendulum+circle+notexture/frame_31.png  (a) Shape mask | ../Python/frame_31.png  (b) Textured shapes |
| Figure 2.1 Synthesised images. | |

We selected three successive synthesised images as shown in Figure 2.2 to illustrate the artificial motion, the instantaneous velocity of the accelerating circle exceeded that of the circle in constant velocity. The simple pendulum is at the centre position between the resting equilibrium position (maximum velocity and zero acceleration) and its limit position (zero velocity and maximum acceleration). The rotating small triangle…

|  |  |  |
| --- | --- | --- |
| ../Python/pendulum+circle+notexture/frame_30.png |  |  |
| Figure 2.2 Selected synthesised image masks for experiment. | | |

The second image sequence involving linear motion is synthesized by using images from the Middlebury database [4]. A subpart of a frame from Mequon (the block of two figures in Figure 2.3) in Middlebury is embedded in a Wooden background. The Mequon sub-frame shifts along a linear trajectory to the lower right corner, both horizontally and vertically.

For evaluating acceleration’s “…” components decomposing algorithm, we use the same images to manipulate motion involving rotation for testing radial and tangential acceleration. The Mequon square rotates at the center of the background to ensure the motion is formed by both radial and tangential acceleration. Figure 2.3 illustrates both linear and circular motion of this set of image sequences, (a) is Mequon at the starting point, after moving to the left and right corner 32 pixels Mequon reach the position in (b). In (c) Mequon rotated 10 degree centred (d) is Mequon rotated 30 degree.

|  |  |
| --- | --- |
| testImg/8pixels-0.png  (a) | testImg/8pixels-3.png  (b) |
| testImg/rotating-10.png  (c) | **testImg/rotating-20.png**  (d) |
| Figure 2.3 Examples of artificial motion. | |

## Real-world Sequence

Apart from the synthetic images sequences, real world data is also important. A pool break video from an overhead view is used to evaluate the performance of acceleration detection in the real world. An overhead view can provide the best view to observe the motion of the objects during movement without normalize or visual angle issue, as shown in Figure 2.4 (a). In order to compare with the experimental data with an estimated value, we use the Hough transform to locate the position of snooker balls and one consistently detected ball along in all frames of the sequence was selected as the test data. The ground truth of selected ball positions was recorded manually.

|  |  |
| --- | --- |
| ../Test%20Img/PoolBreak/09.jpg  (a) Pool break video | ../Python/PoolBreakDetCirc/PoolBreak-09.jpg  (b) Detecting the snooker balls using Hough transform |
| Figure 2.4 Ground truth evaluation data[[2]](#footnote-2) | |

|  |
| --- |
| Screen%20Shot%202015-09-11%20at%2000.37.57.png |
| Figure 2.5 The trajectory and the acceleration of the object |

Figure 2.5 is the zoom-in trajectory and estimated acceleration (arrows) of the selected snooker ball and the positions (red dots) are sampled every three frames for visualisation. The ball begins to move at the left bottom of the image and decelerates until it is hit by another ball at frame 26 (the turning point). The collision changed the trajectory and acceleration of the object. The ball stopped at the top of the image at the end of the sequence. The direction of the arrows denotes the resultant acceleration of the snooker ball at that moment. If the arrow point forward, it denotes the snooker ball is accelerating and vice versa.

## Gait Databases

In Chapter 5 we suggest a new technique to detection heel strike for gait analysis. We evaluate our method on three benchmark databases: SOTON [5], CASIA [6], [7]and OU-ISIR [8]. The gait data used in this paper is collected with various controlled environments. Here we are going to introduce them.

### SOTON

The SOTON gait database [5] was built at 2002 by the University of Southampton.

It

|  |  |
| --- | --- |
|  |  |
|  |  |
| Figure 2.6 Examples of inside data and processed silhouette that we used in this thesis. | |

### CASIA

[6]

|  |  |
| --- | --- |
|  |  |
|  |  |
| Figure 2.6 Examples of inside data and processed silhouette that we used in this thesis. | |

### OSAKA

[7]

|  |  |
| --- | --- |
|  |  |
|  |  |
| Figure 2.6 Examples of inside data and processed silhouette that we used in this thesis. | |

## Conclusion

In this Chapter, we have introduced the synthetic databases that we built to evaluate acceleration. In the next Chapter, we are going to introduce the previous research of Optical Flow.

# Analysing Motion in Image Sequence

## Motion in Image Stream

## Optical Flow

The concept of optical flow was first described by James J. Gibson in 1950. Optical flow denotes the apparent motion between the observer and the observed object caused by relative motion [6]. It has been widely used in many fields of image processing such as motion estimation and video compression. For an image, optical flow is the change of brightness patterns though the image sequence. Figure 3.1 (a), (b) are two successive frames and (c) is the optical flow between them. Thus, we can determine much image information from optical flow: the people and the train are highlighted by optical flow whereas the static objects (e.g. the trees) are not.

|  |  |
| --- | --- |
| ../Python/frame-00002.jpeg   1. Frame N | ../Python/frame-00001.jpeg   1. Frame N+1 |
| 1. The optical flow between two consecutive frames | |
| Figure 3.1 Optical flow between two successive frames | |

## Previous Optical Flow Techniques

Optical flow estimation is one of the earliest and still active research topics in computer vision. In 35 years, many methodological concepts have been introduced and have improved performances gradually [2]. In this section, we will give a brief description of the most representative techniques among different types.

### Differential Method

Horn and Schunck found the first differential approach on computing optical flow in 1981. It represented the beginning of the variational techniques to Computer Vision [1]. They estimated optical flow from the spatiotemporal derivatives of image intensity based on brightness constancy and motion smoothness. Assuming the brightness of point P(*x,y*) at time *t* is E(*x,y,t*), the constraint is that the brightness of point P is constant between successive moment, hence:

the change of brightness can be expanded by Taylor series approximation:

so:

We use u, v to denote , , equation (2.10) can be written as:

There are two unknowns, and , with only one constraint in equation 2.5. Therefore, in order to solve the problem, Horn and Schunck introduced an additional constraint: motion is smooth.

The smoothness constraint assumed that the neighboring pixels of an object have a similar movement tendency. The problem then becomes one of minimizing the change of the optical flow along both the horizontal and vertical directions. The rate of flow change is considered as an error :

we assume the error of equation (2.6) is:

then the total error is:

Optical flow is the solution of the velocities (, ) that minimizes the total error [9].

### Region Based Matching

Block matching is one of the most fundamental method in region based matching techniques. The algorithm assumes that the intensity of every single pixel remains constant between the successive frames if the motion is continuous (there is no occlusion) [2]. Optical flow can be easily computed by determining which block best matches current block in a chosen neighbourhood.

The implementation of region based matching technique can be achieved by minimizing the sum-of-squared difference (SSD) between blocks in the image [10]:

where is the pixel value of position at time t, is the pixel value apart from point , , respectively at frame . The matching block is that where the error is minimum within the search area, and the optical flow change in position between the current block and the matched block [11].

### Dense Optical Flow

Farneback developed a dense optical flow estimation algorithm based on polynomial expansion [12]. In this algorithm, the neighbour of each pixel can be approximated by the polynomial expansion:

Because optical flow is based on the assumption that the image intensity is constant. Therefore, if has moved by :

If is non-singular, the displacement is:

### DeepFlow

DeepFlow is an emerging optical flow technique in recent years due to excellent performance on large displacement estimation and non-rigid matching. It was found by Weinazepfel, Revaud and others in 2013. DeepFlow made a step towards bridging the gap between descriptor matching algorithms with large displacement optical flow techniques [13]. The outline of DeepFlow is shown in Figure 2.2. We will introduce it in two parts: the deep matching algorithm and the energy minimization framework.

|  |
| --- |
| Screen%20Shot%202016-06-07%20at%2014.46.03.png |
| Figure 3.2 The outline of DeepFlow [13] |

#### DeepMatching

The deep matching algorithm first splits the SIFT descriptor from a 128-dimensional real vector into four quadrants: the gradient orientations of the interest point are changed from into where . In order to maximize the similarity between the reference and target descriptor, DeepFlow optimize the positions of on target descriptor rather than keep them fixed:

where is one quadrant of the reference descriptor. By assuming the quadrants are moving independently, a coarse to fine non-rigid matching can be obtained efficiently.

If and denotes the reference and target descriptor respectively, the optimal warping is the one that maximizes the similarity between the pixels:

where returns the position of pixel in . If define recursively then we can obtain the optimal warpings that are largely robust to deformation [13].

#### DeepFlow

DeepFlow is an energy minimization function which is similar with Horn-Schunck. It is based on the same two assumptions: intensity constancy and smooth motion. In addition, DeepFlow blends an extra term, deep matching, into the framework:

where is the weighted data sum, is the smoothness term and a matching term .

A robust penalizer is applied to each term:

with which was determined empirically [14].

The **data term** consists of two penalizers of brightness:

where the first term is the penalizer over image channels, the second one is the penalization for the x and y axes. is the flow we seek to estimate: , is the number of the image channels. is a tensor which is normalized by spatial derivatives:

is the spatial-temporal gradient . is the spatial normalization factor to reduce the impact of small gradient locations and to prevent the factor to be zero. The gradient constancy penaliser is normalized along the x and y axes respectively:

where and are the gradient derivatives respect to the horizontal and vertical axis.

The **smoothness term** in DeepFlow is a penalization for gradient flow:

The purpose of the **matching term** is to find the most similar flow to the known vector as previously introduced in 2.3.1. The difference is estimated by:

Due to the matching being not totally dense, a binary term b(x) is added into the matching term. *b*(*x*) equals 1 if and only if there is a match at position . is a weight that low in flat area. The optical flow we seek to estimate can be obtained by minimising the energy function (2.17) [13].

### Newer Flow

## Previous Acceleration Papers

There has been little work as yet analysing acceleration before we determined gait events through acceleration flow in our first paper [17].  Dong *et al*. [18] differentiated the velocity field without considering an analytic solution for isolating acceleration whereas [19] extended the brightness constraint to three frames and solved the problem in an extremely complex manner.

We are pleased to see emerge acceleration research contemporaneous when I started my PhD, however, they remain area need to improve.

## Experimental Results of Exiting Algorithm on Synthetic and Real Images

We now look at results from computing the flow from the synthetic test images. The resultant images have been normalised for visualisation.

### Experimental Results for Synthesised Images

|  |  |
| --- | --- |
| vx.png  (a) Horizontal velocity | vy.png  (b) Vertical velocity |
| Figure 3.5 Velocity detection by Block-Matching | |

Figure 3.5 illustrates the acceleration detection results of Block-Matching. The instantaneous velocity of the circle with constant velocity is smaller than the background since the velocity of this circle is negative. There are two reasons causing the negative velocity. First, we chose the middle frame as the start frame in order to maintain the same reference position between two instantaneous velocities. Second, the origin of the image is the top-left corner of the image. Therefore, the motion to the right or down will be considered as a positive velocity and vice versa. In addition, the circle with constant velocity did not appear in the vertical component since it has no motion along the vertical axis. The small triangle is brighter in the vertical component than in the horizontal component because it is rotating vertically. The accelerating circle is detected in horizontal and vertical velocity even though it does not have vertical motion. The previous and current position of the objects are all detected in the velocity detection results.

|  |  |
| --- | --- |
| Screen%20Shot%202015-09-02%20at%2011.45.01.png  (a) Horizontal velocity | Screen%20Shot%202015-09-02%20at%2011.44.43.png  (b) Vertical velocity |
| Figure 3.6 Velocity detection by Horn-Schunck | |

In the result of velocity, all the objects are detected by Horn-Schunck. There is not much difference among detected objects even though they have different motions.

|  |  |
| --- | --- |
| Screen%20Shot%202015-09-02%20at%2011.42.58.png  (a) Horizontal velocity | Screen%20Shot%202015-09-02%20at%2011.42.41.png  (b) Vertical velocity |
| Figure 3.7 Velocity detection by Farenback | |

Farneback has very similar results with Figure 2.7 and 2.8, however, the edges of objects in the results are much more blurred.

The stationary shape is not in any of the resulting images because it does not have any motion. There is always some noise around the detected objects. The noise around the object is the part of objects in the previous and next frame which do not overlap with the current objects. This is due to the main limitation of optical flow, assuming that neighbouring pixels have the same movement tendency. It leads to the neglect of edges between different areas moving differently [1].

### Experimental Results for Real-world Sequence

|  |
| --- |
|  |
| Figure 3.10 Acceleration detection results in comparison with estimated value |

Figure 2.10 is the line chart of acceleration detection algorithms compared with estimated value along in all frames of the sequence. Block-Matching has been normalised for comparison. In the chart, estimated value and Block-Matching both illustrate same tendency that the ball was slowing down until it was hit again at frame 26.

Table 2.1 The correlation factor between estimated acceleration and detected results

|  |  |  |  |
| --- | --- | --- | --- |
|  | Horn-Schunck | Block-Matching | Farneback |
| **Correlation factor** (58 frames in total) | 0.0609 | 0.204 | 0.0161 |

Table 2.1 illustrates the correlation factor (what is correlation factor???) between ground truth and the detection results of optical flow algorithms. The ground truth is estimated manually by the object positions detected using the Hough Transform. Of the three techniques, although Block-Matching has the highest correlation with estimated acceleration but it is still a relatively low correlation. There are several reasons might lead to the low figure. First, there exits certain amount of error in the position information detected by the Hough transform due to limitations of the algorithm: it is not robust to lighting condition varies nor to the perceived shape distortion caused by fast motion. On the other hand, the texture of the object is essentially for optical flow to detect motion however the test object has very smooth surface which has big effect on the results.

In addition, the choice of block size and searching area is critical for region based techniques. In the experiment, the selected size maybe not perfect for the motion in this kind of scenario. The static background also gives difficulties to Horn-Schunck to detect motion because of the extra assumption it makes: motion continuity. The correlation factor for Farneback is much less even though it was Block-Matching. This is mainly due to the extra assumption of Farneback: the displacement field varies slowly however the motion in experimental video is opposite.

## Conclusion

We have introduced the optical flow techniques we implemented in this thesis. In the next Chapter we will give a description of the basis of the acceleration decomposition algorithm.

# Theoretic and Implementations on Synthetic Images

## Motivation

Our new approach in this paper retains the elegance of the original Horn Schunck formulation with an approach that isolates only acceleration, thus allowing more detailed analysis of complex motion fields, and we provide a more general experimental analysis.

## Estimation of Acceleration Flow

### Recovering Acceleration Flow from Optical Flow

Acceleration is a vector describing the magnitude and direction of the change of velocity. Average acceleration is the average change rate of velocity respect to time interval. As with velocity, when the time period approaches zero, it is termed instantaneous acceleration:

We follow the earliest optical flow variational approach for estimating optical flow, Horn and Schunck’s work. We extended the initial assumption that when a point moves from first frame to second on the image, the intensity does not change, which is still used by most optical flow algorithm, is true during three frames. If we expand the intensity of a point on image pattern at time into the future by using a Taylor series:

Similarly, for extension into the previous time:

where contains higher order terms. If we add (4.2) and (4.3):

If expand equation (4.2) by Taylor expansion and ignore the higher order terms:

where the higher order terms are ignored. Dividing (2) by, the gradient constraint is yielded:

where , and consists of horizontal and vertical components .

If the acceleration is dynamic from frame to frame, then this become an ill-posed problem. More commonly, motion is smooth which means acceleration is usually constant during a small period. Here, we assume that the acceleration does not change during three consecutive frames. Then velocity can be presented by from Newton’s laws:

where denotes the velocity vector at time and the acceleration vector is composed by horizontal and vertical components . Differentiating (4) with respect to time, we can obtain the Optical Flow Constraint Equation (OFCE) of acceleration:

where indicates the second order of image intensity respect time.

Acceleration also has similar smoothness characteristics to velocity in that neighbouring pixels tend to have similar acceleration. This shows a natural linkage between velocity and acceleration analysis in image sequences. By following a similar solution to Horn Schunck, we can determine the acceleration flow in images. We now have the basis for detecting acceleration, we shall now move to evaluating this approach to determine whether we can indeed detect acceleration from image intensity.

Continue algorithm…

(3.1)

If we follow the original optical flow variational approach, we can expand an image at time into the future by using a Taylor series as:

(3.2)

Similarly, for extension into the previous time:

(3.3)

where contains higher order terms. If we add (3.2) and (3.3):

(3.4)

and take the limit as then we can ignore as it also tends to zero which leaves:

(3.5)1

Since and the other one we can substitute for to give:

(3.6)2

which contains all the motion in the images, including acceleration. In the original optical flow developments, the term:

(3.7)

can be reformulated to form an optical flow equation. For acceleration (differencing two frames rather than one), the process has led to a similar equation but with increased complexity due to second-order partially differential terms such as. Though it has not led to an analytic solution for determining acceleration, it does however suggest that differencing two frames rather than one can lead to images that embed the acceleration.

### Analysing Acceleration on Synthetic Image Sequences

We evaluate our new approach first on synthetic images to assess performance before analysis on real images to show application capability. The advantage of synthetic images is that the input signal is without noise, specularity, or other types of noise. Also, the motion field and scene properties can be manipulated as required. This image sequence involving linear motion is synthesized by using images from the Middlebury database [4]. A subpart of a frame from Mequon (the block of two faces in Fig. 3 (a)) in Middlebury is embedded in a frame from the Wooden images. The Mequon sub-frame shifts along a linear trajectory to the lower right corner at speed 1/4 pixel/frame, both on horizontal and vertical axes. Sub-pixel motion is achieved by down-sampling the images.

|  |  |  |
| --- | --- | --- |
| woodenMequon.png(a) Synthetic image | noAcc.png(b) Acceleration field when Mequon is undergoing non-acceleration motion. | acc.png(c) Acceleration field when Mequon is accelerating. |
| Figure 4.1 The detection results of synthetic images | | |

Fig. 3. (a) shows one example frame of the synthetic sequence, (b) and (c) shows the acceleration detection results under the motion without and with acceleration respectively. When Mequon moves without acceleration, there is little acceleration flow detected, except random noise. Encouragingly, as opposed to the uniform motion the new algorithm detected evenly distributed acceleration flow in the right-hand frame. We would have liked to compare performance with the other acceleration technique [15] but the implementation is unavailable for that complex technique whereas [4] lacks the analytical basis here.

|  |
| --- |
|  |
| Figure 4.2 The experimental result on Yosemite. |

We also evaluated our new algorithm on the Yosemite image sequence which is a challenge for fundamental optical flow algorithms since the velocities of different areas vary and the edges are occluded between the mountains. The upper right corner translates to the right with a speed of 2 pixels/frame and the speed in the lower left area is about 4 to 5 pixels/frame [10]. The non-uniform motion is caused by the asymmetrical We also evaluated our new algorithm on the Yosemite image sequence which is a challenge for fundamental optical flow algorithms since the velocities of different areas vary and the edges are occluded between the mountains. The upper right corner translates to the right with a speed of 2 pixels/frame and the speed in the lower left area is about 4 to 5 pixels/frame [10]. The non-uniform motion is caused by the asymmetrical projection of 3-D motion onto the 2-D image surface. Our acceleration measurement produces a poor result as shown in Fig. 4 since Horn and Schunck’s algorithm assumes global smoothness and sub-pixel motion which is violated in this case.

Table . Error analysis of acceleration estimation algorithm

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Data AE | ***Army*** | ***Mequon*** | ***Schefflera*** | ***Backyard*** | ***Dumptruck*** | ***Grove2*** | ***RubberWhale*** | ***Walking*** |
| Average (°) | 10.84 | 28.3 | 33.89 | 17.82 | 12.36 | 14.31 | 9.74 | 17.82 |
| SD (°) | 8.61 | 20.18 | 19.77 | 21.59 | 17.98 | 13.82 | 8.77 | 19.54 |

For more precise, TABLE 4.1 reports the performance measures of our acceleration estimation algorithm on Middlebury optical flow benchmark data. We computed the average Angular Error (AE) and Standard Deviation (SD) between the experimental results and pseudo ground truth. Since the ground truth flow can be accessed from the Middlebury optical flow database is only between two frames in the dataset, we use the acceleration flow estimated based on MDP-Flow2 [16] by (6) as the pseudo ground truth. MDP-Flow2 is a highly-rated optical flow estimation algorithm on the Middleburry evaluation website[[3]](#footnote-3). The results in Table I show that the new acceleration estimation algorithm is not accurate enough to be used as a motion descriptor because of the stringent constraints. Fig. 5. shows the colour map of the pseudo ground truth and estimated acceleration flow of data “Backyard” and “Walking”. In the pseudo ground truth, the falling ball in (c) and the walking legs in (e) has more acceleration than other motion areas. The acceleration field illustrates detection of features consistent with acceleration features, which velocity analysis lacks.

|  |  |  |
| --- | --- | --- |
| ../../Downloads/eval-data/Backyard/frame10.png  (a) Backyard | (b) Backyard pseudo ground truth | (c) Backyard acceleration |
| ../../Downloads/other-data/Walking/frame10.png  (d) Walking | (e) Walking pseudo ground truth | (f) Walking acceleration |
| Figure 4.3 The colour map of acceleration flow: (b), (e) are pseudo ground truth [7] and (c), (f) detection by the new approach on the right | | |

### Estimating Motion Flow via Other Flow Estimation Methods

Since the motion in real images is often large we want to seek more generalized form for recovering acceleration from image sequences, we opt to use the state of the art optical flow estimation algorithm to approximate the acceleration than to use the new algorithm. In this paper, we use DeepFlow [13], which is a popular new technique with excellent performance for large displacement estimation and non-rigid matching as our fundamental technique of our algorithm.

As in our previous work [17], we approximate the acceleration by differencing the velocity field between frames. By reference to the same starting position, the time axis is reversed when estimating the previous velocity :

Inspired by the analysing in 4.2.1 and discreteness of computer images, acceleration can be determined by computing the difference in optical flow between successive frames. We can consider the moving distance of the pixels that are estimated by optical flow as velocity of the pixel. The unit is pixels per frame. Then acceleration can be derived by the difference of velocities. In order to maintain the consistency of the object position in the implementation, we choose the middle frame as the reference. In other words, the velocities between adjacent frames are computed respectively by optical flow backwards and forwards using the middle frame as the initial moment. Therefore, the accelerations along both axes can be obtained by:

(3.8)

### Analysing New Algorithm on Synthetic Images

|  |  |
| --- | --- |
| (a) Moving with constant velocity | (b) Moving with acceleration |
| Figure 4.4 The experiment results on Yosemite | |

### Tangential and Radial Acceleration

Acceleration for motion is composed of two components: tangential and radial acceleration. The tangential component changes the magnitude of the velocity vector and the direction is located in the tangent line of the trajectory (increasing or decreasing the speed). The radial component (also called centripetal acceleration in circular motion) changes the direction of the velocity and it points to the centre of the curved path (normal to the direction of velocity), as shown in Fig. 8. Motion is composed of linear or circular motion; therefore, the motion incorporated in images is either linear or circular if the time interval is sufficiently small.

Acceleration for curved motion is composed of two components: tangential and radial acceleration. The tangential component changes the magnitude of the velocity vector and the direction is located in the tangent line of the trajectory (increasing or decreasing the speed). The radial component (also called centripetal acceleration in circular motion) changes the direction of the velocity and it points to the centre of the curved path (normal to the direction of velocity) [Hibbeler, 2009], as shown in Figure 3.1.

|  |
| --- |
| ../../Desktop/Tangential-acceleration.PNG |
| Figure 4.5 The relationship between resultant acceleration, tangential acceleration and radial acceleration[[4]](#footnote-4) |

We assume that the pixels which following a circular motion rotate along the same arc in any three consecutive frames because three points can determine one and only one circle. The centre of the circle can be calculated by the positions of the pixel in the consecutive frames. Connecting these three points with straight lines and applying perpendicular bisectors to them, the centre of the circle is then located at the intersection of the perpendicular bisectors of the straight lines, as shown in Figure 3.2.

|  |
| --- |
| ../../Users/user/Downloads/circleCente |
| Figure 4.6 Location of radial acceleration centre |

Suppose the coordinates of the points are: , are the perpendicular bisectors of and , hence:

then the coordinate of the circle centre can be obtained by:

By geometry, the tangential acceleration and radial acceleration can be estimated by:

where , , is the angle between and horizontal axis.

|  |
| --- |
| CVPR/Screen%20Shot%202016-11-04%20at%2000.17.23.png |
| Figure 4.7 The acceleration of a particle along arbitrary curved path is composed of radial and tangential component. |

We now have the basis for detecting acceleration and its extension to a more generalised form. We shall now move to evaluating these approaches to determine whether we can indeed detect acceleration from image intensities.

### The Synthetic Images and Error Analysis of SAD

* Where are they from
* How they been made
* Ground truth

We evaluate the decomposed approach first on synthetic images to assess performance before analysis on real images to show capability in real applications.

|  |  |
| --- | --- |
| ax.png   1. Horizontal acceleration | ay.png   1. Vertical acceleration |
| Screen%20Shot%202015-09-02%20at%2012.25.08.png   1. Magnitude of acceleration | |
| Figure 4.8 Acceleration detected by Block-Matching | |

In the acceleration results, the circle with uniform motion is barely being detected however the small triangle is still detected. The accelerating circle and the simple pendulum are much brighter at their middle position. The simple pendulum has more acceleration at the bottom than the top. This is mainly because optical flow detects motion depending on texture so the narrow top is barely detected.

|  |  |
| --- | --- |
| acc_x_HS.png   1. Horizontal acceleration | acc_y_HS.png   1. Vertical acceleration |
| Screen%20Shot%202015-08-20%20at%2021.55.58.png   1. Magnitude of acceleration | |
| Figure 4.9 Acceleration detection by Horn-Schunck | |

In the Figure 2.8 (c), the magnitude of the accelerating objects is much more obvious in the current position than the Block-Matching result. However, the objects with constant velocity are also being detected by Horn-Schunck.

|  |  |
| --- | --- |
| FB_acc_x.png   1. Horizontal acceleration | FB_acc_x.png   1. Vertical acceleration |
| FB_acc.png   1. Magnitude of acceleration | |
| Figure 4.10 Acceleration detection by Farneback | |

The Mequon sub-frame shifts along a linear trajectory to the lower right corner at speed 1/4 pixel/frame, both on horizontal and vertical axes. Sub-pixel motion is achieved by down-sampling the images

|  |  |  |
| --- | --- | --- |
|  | Linear motion without acceleration | Linear motion with acceleration |
| Velocity | ../../../Users/user/Desktop/shift_noAcc | ../../../Users/user/Desktop/shift_Acc |
| Resultant Acceleration | ../../../Users/user/Desktop/shift_noAcc | ../../../Users/user/Desktop/shift_Acc |
| Radial Acceleration | ../../../Users/user/Desktop/shift_noAcc | ../../../Users/user/Desktop/shift_Acc |
| Tangential Acceleration | ../../../Users/user/Desktop/shift_noAcc | ../../../Users/user/Desktop/shift_Acc |
|  |  | |

|  |  |  |
| --- | --- | --- |
|  | Circular motion without angular acceleration | Circular motion with angular acceleration |
| Velocity | vel.png | ../../../Users/user/Desktop/rotation_Acc |
| Resultant Acceleration | acc.png | ../../../Users/user/Desktop/rotation_Acc |
| Radial Acceleration | rad.png | ../../../Users/user/Desktop/rotation_Acc |
| Tangential Acceleration | tan.png | ../../../Users/user/Desktop/rotation_Acc |
|  |  | |

## Estimating Acceleration via Other Flow Estimation

# Detecting Acceleration for Gait and Crime Scene Analysis

## Gait and Heel Strike

## Heel Strike Detection via Radial Acceleration

### Key Frame Detection

### Positioning and Verification

## Experiments

### Key Frames Positioning

### Performance on Different Databases

### Error Analysis

# Higher Order Motion Flow

# Conclusion and Future Work

## Conclusion

Apart from identifying behaviours, acceleration also offers an alternative approach to understand image content. For examples in autonomous driving, the acceleration of other cars can help a computer to make correct decisions about other vehicles’ motion trajectories.

## Future Work

### Behaviour Recognition

### Scene Segmentation

Summary on database

Conclusion on other chaps

Four main contributions:

* Acceleration
* Evaluation
* Gait and evaluation
* Generalization

1. Your first appendix
2. Your second appendix

Glossary of Terms

List of References

Bibliography

1. Images are taken from: <http://news.stanford.edu/2014/04/24/walking-vs-sitting-042414/> and <http://www.wisegeek.com/what-are-the-different-types-of-track-spikes.htm#man-running-on-stret> [↑](#footnote-ref-1)
2. Video is taken from: <https://www.youtube.com/watch?v=XpPbrLiSMCY> [↑](#footnote-ref-2)
3. http://vision.middlebury.edu/flow/eval/ [↑](#footnote-ref-3)
4. Image is taken from: <http://physics.tutorvista.com/motion/tangential-acceleration.html> [↑](#footnote-ref-4)