Project Plan

Wolfgang Baumgartner

Thursday 4th May, 2017

Summary

[todo] Write a summary

Samenvatting

Dit is de samenvatting.

Het zou kunnen beschrijven hoe handig het is dat de template gelijk de documentatie is, maar helaas wordt dit dus niet gedaan…

Preface

This project plan should contain the following:

- Introduction
- Literature study
- Feasability study
- timeline for production phase
- deciding about scope of this report
- Should this report be a complete document that makes clear to the outsider what the background, intention and outline of this project?

Contents

1 Introduction

1.1 Ideas

[rro] Ideas for this research project

[todo] ask RAM members

[todo] vragen over Linux distributie

- Generic stereo vision module
- Solution to a specific problem
  - Object recognition
  - Drone orientation and localisation
  - Monitoring of a process
  - Supporting system for sports referee
  - In combination with SLAM
  - Control of 3D-printed objects
  - Stereo vision camera as input for feedback control loop
  - hand gesture control
  - facial recognition
  - 3D reconstruction
  - generate 3D model for printer
  - research on how to use stereo vision as input for control loop like SHERPA arm or drone(position of camera, algorithms, fps)

1.2 Questions

- How relevant is stereo vision in a research context?
- How relevant is stereo vision for RaM?
- What is the topic of ongoing research in stereo vision?
- What kind of problems are solvable with stereo vision?
- How much literature do I have to read to know what the current state of research is?
- Would it be interesting to publish an article presenting the results?

1.3 People to talk to

- Ferdi van der Hejiden, RaM, image processing
- Han Wopereis, RaM, Aeroworks
- Klaas Jan Russcher, RaM, Rove
- Eamon Barrett, RaM, Sherpa (maybe not necessary, already talked with other sherpa member)
- Bert Molenkamp, CAES

  [done] afspraak met Bert

- Johan Engelen, RaM, Inspectie robots

  [done] afspraak met Johan

- Stefano Stramigioli, RaM, all projects

- Douwe Dresscher, RaM, ROSE

1.4 Context

[todo] Read more literature

[todo] Research capabilities of RaMStix

[rro] context for stereo vision and posible application

A stereo vision module could be interesting for the SHERPA project. The rover has an arm that needs to pick up a small quadcopter to change its battery. Stereo vision could be used to detect the drone without the necessity for markers on the drone. Additionally, it would be possible to track the distance between arm and drone. A first estimation of requirements would be to have a framerate between 10 and 30 to detect a drone in (slow) motion. The used cameras should be fitted with a global shutter to get pictures with sufficient quality. A problem could be missing computation power of the RaMStix when more sophisticated image processing is necessary. For a stereo vision implementation in general, the existing bus could be a bottle neck.

1.5 Research question

[todo] decide on what to do

[rro] specific research or problem I want to solve

[rro] how efficient/effective is my solution?

[rro] how appropriate is my solution?

1.6 Objective

[rro] desired outcome

1.7 Approach

[rro] how to solve problem

1.8 Report outline

[rro] outline of this document

2 Background

2.1 How does stereo vision work?

[rro] briefly explain stereo vision

2.1.1 Rectification

[rro] explain preprocessing step

[rro] includes calibration for epipolar geometry

2.1.2 Pixel matching

[rro] algorithm to find the corresponding pixels

2.1.3 Comparison

[rro] extract depth information

[rro] plot disparity map

2.2 How does random application work?

[todo] decide application or research question

[rro] background for chosen problem

2.3 What is a RaMstix?

[rro] characteristics of RaMStix relevant for this project

3 Approach

[rro] feasibility study/design space exploration

[todo] propose specific ways of implementing stereo vision

4 Planning

[rro] Gantt chart

5 Combining hardware and software for stereo vision implementation

5.1 Abstract

[rro] Performance comparison

Build a stereo vision module that makes use of both the FPGA and the Gumstix, that means an implementation that combines hardware and software in an appropriate way. Compare performance with other implementations that only use hardware(FPGA) or software.

5.2 Background

[todo] find reference values

[todo] decide division hardware/software

[todo] find literature with similar approach

[todo] Measure performance

6 DVS

6.1 Abstract

The Dynamic Vision Sensor as used in Schraml et al. (2010) and Belbachir (2010) is an asynchronous sensor that detects changes in intensity over time and sends the pixel address as event. As there is no similar sensor available, the Ramstix is used to extract similar events from a stereo camera. This reduces the amount of data that needs to be processed. A disparity map can be calculated from the extracted events with much lower resource requirements than for typical stereo vision which makes this approach perfect for an embedded systems project.

6.2 Background

The above mentioned sensor consists of an array of 128*128 pixels which detect intensity changes over time. Every pixel works asynchronously and can send two different events signaling an increase or decrease in intensity at a certain position. Each sent event consists of the event polarity and the pixel position. The asynchronous output leads to a high temporal resolution while the amount of data compared to a whole frame from a typical camera is notably decreased. Using this sensor for stereo vision like in Schraml et al. (2010) makes for a quite efficient embedded system.

The outline of the stereo vision algorithm in Schraml et al. (2010) can be seen in figure 6.1. The asynchronous events enter the pre-processing stage which is a preparation for the stereo matching. To make the search of corresponding pixels easier, a transformation is performed which results in rectified images conforming to an epipolar geometry. An event and its matching partner can now be searched in the same horizontal line. The arriving, rectified events are bundled into time periods similar to frames and the stream is put into matrizes.

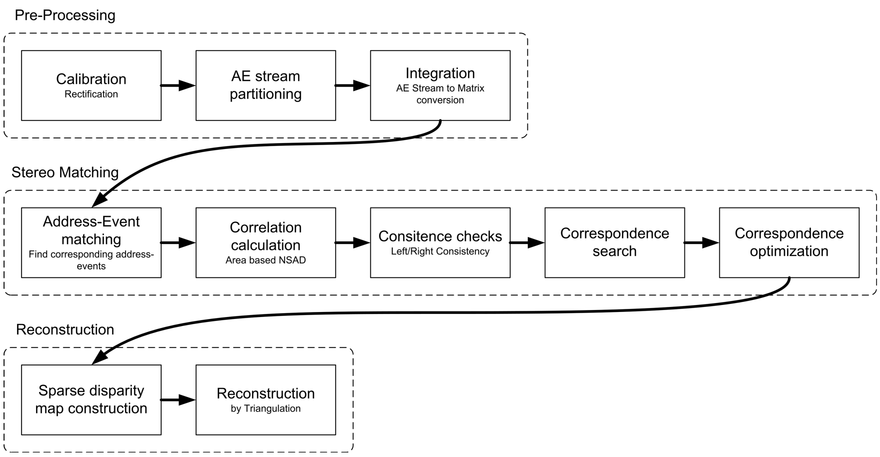

Figure 6.1: A block diagram that shows the stereo matching algorithm in Schraml et al. (2010)

In the stereo matching phase, potential matches are searched and the matching cost for all possibles matches is calculated. This is done from both sides to assure constistency. Actually matching events are found by minimizing the earlier calculated cost.

The distance between matching pixels is put into a disparity map which shows all found events with corresponding distances.

6.3 Approach

On the basis of DVS and the mentioned stereo matching algorithm, a system can be designed that enables efficient object tracking in three dimension with limited resources. Events can be extracted from a stream of images by comparing consecutive images and remembering pixel positions with changing intensity. The resulting events can be processed in the same way as describe above. A usual image sensor already delivers whole frames which means that there is no need for the stream partitioning step. The rest of the steps don't depend on the choice of the sensor.

Research can be done for the correlation calculation block as there are several algorithms for this step with different characteristics. Consistency may or may not be necessary and there may exist less resource-intensive ways to ensure consistent results.

In order to demonstrate the capabilities of the system, the disparity map can be used to track an object and measure its path. Accuracy of the path displays the effectiveness of the system while resource usage gives a measure of the efficiency of the designed system.

A Appendix 1

Tip: Make a copy of this document, since this is a manual of the template. This will prevent losing the document while modifying it. (And probably breaking it.)

A.1 Required TexLive packages

The following packages are required for TeX Live 2012 and 2013 (tested under Ubuntu)

```
texlive
texlive-lang-dutch     -> Dutch hypenation
                          (old TeX Live distributions, 2013-)
texlive-lang-europeans -> Dutch hypenation (amongst others)
                          (new Tex Live distributions 2013+)
texlive-latex-extra
texlive-fonts-extra    -> fourier
texlive-humanities     -> lineno
texlive-pstricks       -> pstricks (2013+?)
                          (not really used in the template??)
```

Bibliography

Belbachir, A. N. (Ed.) (2010), *Smart cameras*, Springer Verlag.

Schraml, S., A. N. Belbachir, N. Milosevic and P. Schön (2010), Dynamic Stereo Vision System for Real-Time Tracking, *Circuits and Systems (ISCAS), Proceedings of 2010 IEEE International Symposium.*