Note: For best view of equations use "GitHub with MathJax" extention for your browser.

FeatureTrackingForAutonomousVehicle

Sequential frame feature tracking for autonomous vehicle

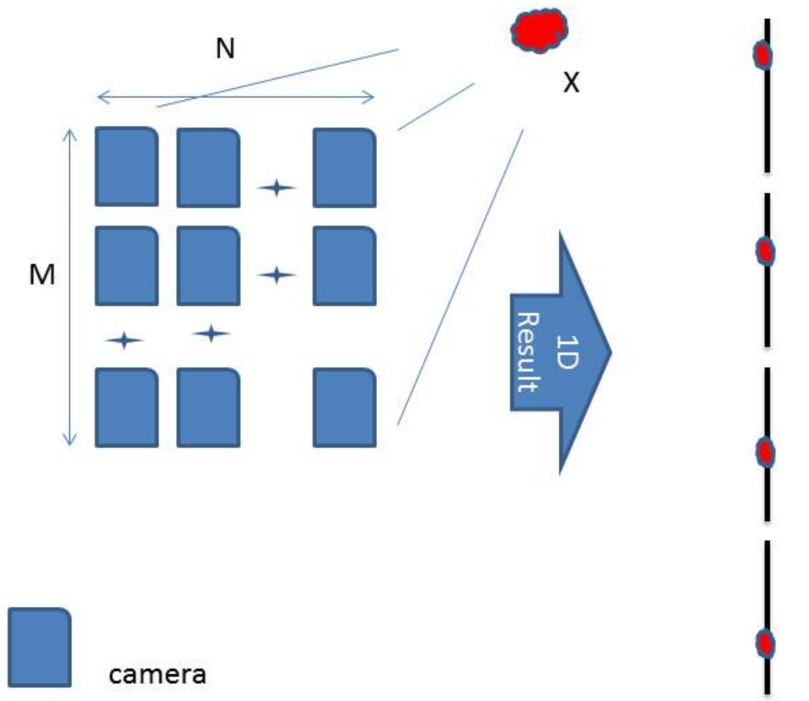
```
#image_processing, #image_geometry, #autonomous_vehicle, #depth_estimation, #camera_array,
#camera_Rig
```

Introduction

This method helps feature detection algorithm. The Extended formula shows that changing vehicle's speed can help to track feature in some region in the image. Now, this solution uses for stereo tracking, but in our framework, we can apply vehicle speed in the sequence of any frames which taken from the moving vehicle. An autonomous car is our goal which has this type of motion.

We can apply this method to many sections of computer vision. In this subsection, we introduce how this model uses and how we can apply this algorithm.

An array camera [2] and image formation of it shown in figure 1. In this situation shows that when an object such X, place in front of the array camera, the images of objects (right images) in this array of camera is **periodic**.



- figure 1: One sample of camera array and image formation

Now, if a sequence of images from an object formed. what is the relation between image position and object position? In figure 2 a camera move with constant speed. The images from object X has been showed on any of camera sequence. With simple assumption, we proof with constant speed, but we proved the general case with a different velocity. This paper proves a theory which relates to the difference of feature position in subsequent images.

We will show in the image sequence of the camera, the features of the image has constant distance on image array(right picture), if the distance of camera's center is less than object distance ($D >> nd$).

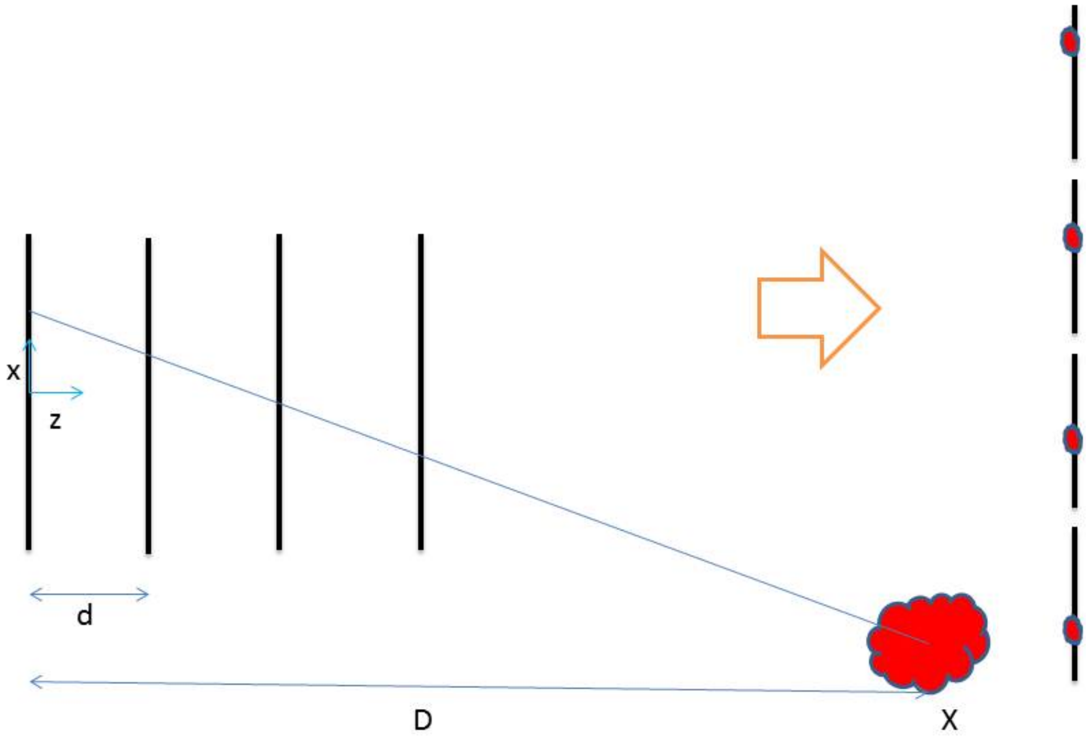


figure 2:Camera in motion image formation

figure 3 : Feature detection in sequence of images. From left to right, 4 consecutive frames have been shown. on top shown which feature track

Difference between array of cameras and sequence of a camera in space

The extracted formula has a major difference with an array of cameras. One of the differences between these methods is the final formula. In array camera, The output relay on depth of the feature. any feature with same depth has a periodic number, **but in our case the output based on the depth and initial position of feature**. Because of this difference the first has big impact on optical imaging but the later could not solve in optical imaging. The second difference is the array of flat can be implemented with many cameras on chip or any surface, but our case just happen when a camera penetrate in space. The implementation of the sequence of the camera is not reported yet.

Applications

1. **Object depth estimation**

The formula in this method can be solved for scale estimation. For example, knowing the speed of the vehicle and three sequences of the frame from it, get the scale factor. This scale shows the object which the features relate to that in a depth of Alfa proportion to camera frame distance.

2. **Warning system for autonomous car object on-road detection**
3. **Feature tracking accelerator**

   This algorithm shows that, when a camera penetrate in a world with fix speed and there is no barrier in the road, then distance of features in two frames must be equal. This algorithm says that we can find a feature within a frame with a search within determining steps.

4. **Classification of forwarding motion and unknown motion**
5. **Scale estimation in vision section**

   A known method of vision solves the problem up to scale. A method like 8-point algorithm just solves up to scale. In that paper, The author introduces a section which helps others to solve the scale with other sensors or assumptions. Our introduced method solves this problem with knowing the speed. Speed is a simple parameter used in autonomous car.

6. **Sub-pixel matching improvement**

   Many geometric equations in computer vision rely on the noise of features position in images. For instance, mankind can match two images with the precision of one pixel, but some algorithm like sift\cite{sift} present sub-pixel matching. In many conditions, some edges in image blurred and determining the position of the corner is not straight forward. Our algorithm can be used in feature tracking where some corner of consequence frame is occluded.

Equation

General Equation

In this paper we used equation P as a point with depth of D(all parameter shows in Fig2)

$$P = \begin{pmatrix} X \\ Y \\ D \end{pmatrix}$$

For the image of point on sequence of images, write such equation(where p is image of P):

$$p_0 = \begin{pmatrix} \frac{X}{D} \\ \frac{Y}{D} \end{pmatrix}, p_1 = \begin{pmatrix} \frac{X}{D+d} \\ \frac{Y}{D+d} \end{pmatrix}, \ldots, p_n = \begin{pmatrix} \frac{X}{D+nd} \\ \frac{Y}{D+nd} \end{pmatrix}$$

If $D >> nd$ or the distance of camera zero to N should be small about the total depth of blocks in Images, then we can write:

$$with$$

$$p_n = \begin{pmatrix} \frac{X}{D} * \frac{1}{(1+\frac{nd}{D})} \\ \frac{Y}{D} * \frac{1}{(1+\frac{nd}{D})} \end{pmatrix}, p_n = \begin{pmatrix} \frac{X}{D} * (1+\frac{nd}{D})^{-1} \\ \frac{Y}{D} * (1+\frac{nd}{D})^{-1} \end{pmatrix}$$

With **Binomial approximation**, we can rewrite the equation as follows:

$$\approx p_n = \begin{pmatrix} \frac{X}{D} * (1-\frac{nd}{D}) \\ \frac{Y}{D} * (1-\frac{nd}{D}) \end{pmatrix} = \begin{pmatrix} p_{0x} * (1-\frac{nd}{D}) \\ p_{0y} * (1-\frac{nd}{D}) \end{pmatrix}$$

In last step, the equation shows that difference in two feature's position should be constant and equal to the $p_{0y}\frac{d}{D}$.

$$\approx p_{nx} - p_{(n+1)x} = p_{0x} * (1-\frac{nd}{D}) - p_{0x} * (1-\frac{nd}{D}-\frac{d}{D}) = p_{0x}\frac{d}{D}$$

$$\approx p_{ny} - p_{(n+1)y} == p_{0y}\frac{d}{D}$$

Extended method

In this section, generalized view of the algorithm shown. When the camera **penetrate with variable speed** in this condition distance of two subsequent frame is not equal. In equation Ex_elementary has shown a basic formula of the variable distance between cameras.

$$p_0 = \begin{pmatrix} \frac{X}{D} \\ \frac{Y}{D} \end{pmatrix}, p_1 = \begin{pmatrix} \frac{X}{D+d_1} \\ \frac{Y}{D+d_1} \end{pmatrix}, p_2 = \begin{pmatrix} \frac{X}{D+d_1+d_2} \\ \frac{Y}{D+d_1+d_2} \end{pmatrix}$$

Ex_elementary

Using Binomial approximation, same as equation Binomial approximation we this equation for variable speed cameras.

$$p_1 \approx \begin{pmatrix} \frac{X}{D} * (1-\frac{d_1}{D}) \\ \frac{Y}{D} * (1-\frac{d_1}{D}) \end{pmatrix}, p_2 \approx \begin{pmatrix} \frac{X}{D} * (1-\frac{d_1+d_2}{D}) \\ \frac{Y}{D} * (1-\frac{d_1+d_2}{D}) \end{pmatrix}$$

Proportion of difference in two consequence is shown in equation Eq_final. This shows the proportion is constant for this condition. For any point in space, this equation uses the relative depth of that point.

$$\frac{p_0 - p_2}{p_0 - p_1} = \frac{d_1 + d_2}{d1}$$

Eq_final

Equation when calibration present

$$P = \begin{pmatrix} X \\ Y \\ D \end{pmatrix}, K = \begin{pmatrix} f_x & \theta & O_x \\ 0 & f_y & O_y \\ 0 & 0 & 1 \end{pmatrix}$$

$$p_{nx} - p_{(n+1)x} = (p_{0x} - O_x)\frac{d}{D}$$

Feature distance offset is same and equal to $(p_{0x} - O_x)\frac{d}{D}$

Result

Problems

Synthetic dataset generator, help to check the algorithm. in this paper we use Vladimir Haltakov dataset generator[1]. We compare the difference for features in two consecutive frames.
For first feature we have:

.	feature in image 1	feature in image 2	feature in image 3	feature in image 4
x	298	296	294	291
y	205	204	202	201

For second feature we have:

.	feature in image 1	feature in image 2	feature in image 3	feature in image 4
x	465	463	470	478
y	110	102	93	85

For third feature we have:

.	feature in image 1	feature in image 2	feature in image 3	feature in image 4
x	159	149	139	128
y	216	216	214	213

When camera frame rate is high enough, the frame is taken with centimeter or millimeter in location. In this configuration we know every cars dashboard and head rig is one meter long, this shows that this algorithm is good for many situation where the vehicle travel. In some condition where the camera has been placed on front light we has problem with depth of point in front of camera can generate violation in algorithm. the bad point in images is like in picture $18_0 0007$ where the line of road is near the camera.

.	feature in image 1	feature in image 2	feature in image 3	feature in image 4
x	217	211	204	190
y	370	373	375	384

Test sample.

Reference

[1] Vladimir Haltakov, Christian Unger, and Slobodan Ilic. "Framework for generation of synthetic ground truth data for driver assistance applications". In: GCPR. 2013.

[2] Ren Ng et al. "Light field photography with a hand-held plenoptic camera". In: Computer Science Technical Report CSTR 2.11 (2005), pp. 1-11.

Cite this if useful!

```
@MastersThesis{MahdiKarimian:Thesis:2013,
    author      =     {Mahdi Karimianshahri},
    title       =     {Cooperative camera localization},
    school      =     {Sharif University of Technology},
    year        =     {2013},
    }
```