Detection of swapped views in stereo image

Alexey Shestov, Alexander Voronov, Dmitriy Vatolin
Department of Computational Mathematics and Cybernetics
Moscow State University, Moscow, Russia
{ashestov, avoronov, dmitriy}@graphics.cs.msu.ru

Abstract

An algorithm for automatic swapped views detection is proposed. No analogues in literature were found for this problem solution. It is based on occlusion detection and motion vectors histogram. The algorithm was tested on 780 frames from 13 movies. The recall\precision diagrams were constructed using two parameters. The complexity is estimated. The drawbacks are analyzed and further directions are proposed.

Keywords: *Image Processing, Stereo Vision, Swapped Views, Channel Mismatch*

1. INTRODUCTION

During a film production in some scenes left and right views can be occasionally swapped. Our goal is to reveal such scenes in the movies. Actually, this type of artifact is not easily detectable by human eye from the first look, because when you see such scene, you understand that something is wrong, but can't understand what exactly is. But correctness of views arrangement can be checked on the basis of foreground-background segmentation and inter-view optical flow analysis.

Currently we have performed early stage research and implemented initial version of the algorithm for swapped views detection.

2. THE MAIN IDEAS

2.1 The necessary definitions

Binocular disparity refers to the difference in image location of an object seen by the left and right eyes, resulting from the eyes' horizontal separation. We will call it **disparity** further.

To estimate disparity we used the **Optical Flow** (or **OF**) algorithm described in the paper[4]. We treated disparity as motion vectors between views. We will use words "**motion vectors**" and "**disparity**" as synonyms.

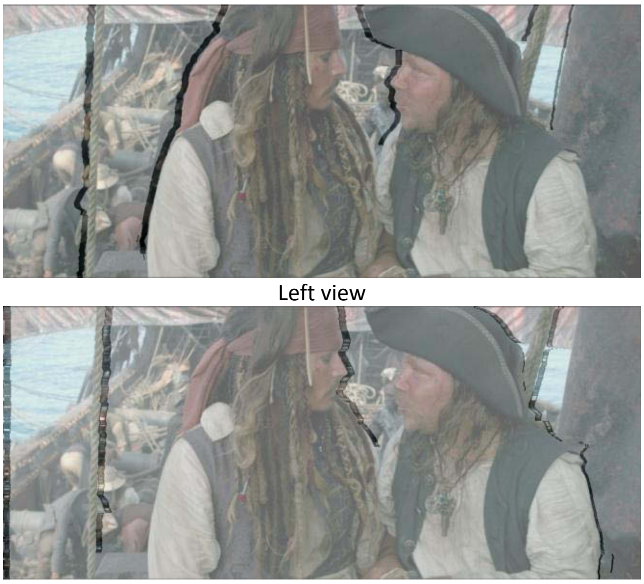
By **occlusions** we mean regions which are presented in only one view of a stereoimage.

Left-right consistency (or **LRC**) is the confidence measure for optical flow [2]. We take the vector A in the point X in the left image, then we take the vector B in the point A+X in the right image. In ideal situation B+A should be equal to zero. So the greater is B+A – the less confident is the vector in the point X.

2.2 Ideas

Two main ideas, which are used in our algorithm:

- The first idea is based on the fact that in the left view occlusions are always leftwards the object and in the right view occlusions are always rightwards the object (see Figure 1).
- The second idea uses the fact that negative parallax regions are one-third of the zone of stereo comfort perception, and positive parallax regions are two-third of the zone of stereo comfort perception, so by an analysis of disparity histogram we can say which disparity values correspond to negative parallax and which values correspond to positive parallax (see Figure 3).



Left view

Right view

Figure 1. Here you can see, that on the left view occlusions are always leftwards the object, and on the right view – rightwards. The frame is taken from the movie "Pirates of the Caribbean: On Stranger Tides".

An illustration of the firsrt idea is presented in Figure 2:

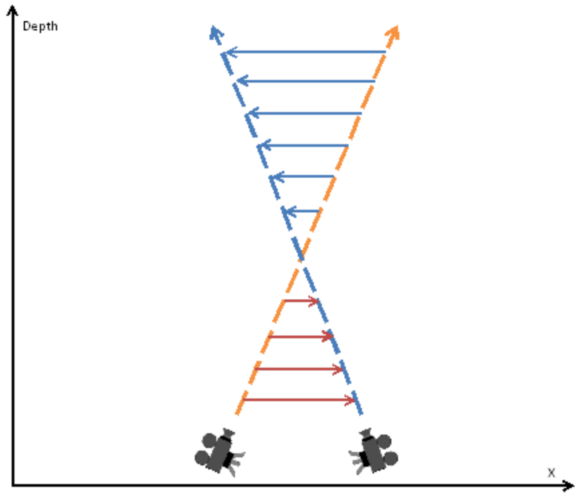


Figure 2. Here rays from the left and right cameras and the motion vector field from the right view to the left view are shown. You can see that disparity is linearly dependent on the depth. On the right view disparity value of an object always is larger than disparity value of background. So,

objects in the right view will always have occlusions rightwards (and in the left view - leftwards).

Disparity is linearly dependent on the depth. In the right view disparity value of an object always is larger than disparity value of background. So, objects in the right view will always have occlusions rightwards (and on the left view – leftwards).

So, we can use some edge detector, then count where are more image edges, rightwards or leftwards the occlusions, and calculate a probability that a current view is left or right.

But there are some frames, where occlusion are so thin, that they are not detected by our algorithm, or if they are detected, we can't say if edges are rightwards or leftwards them. In such cases we use the next considerations: if occlusions are thin, then there mustn't be foreground objects, which depth is much different from background depth. We know that negative parallax regions are one-third of the zone of stereo comfort perception, and positive parallax regions are two-third of the zone of stereo comfort perception (see Figure 3).

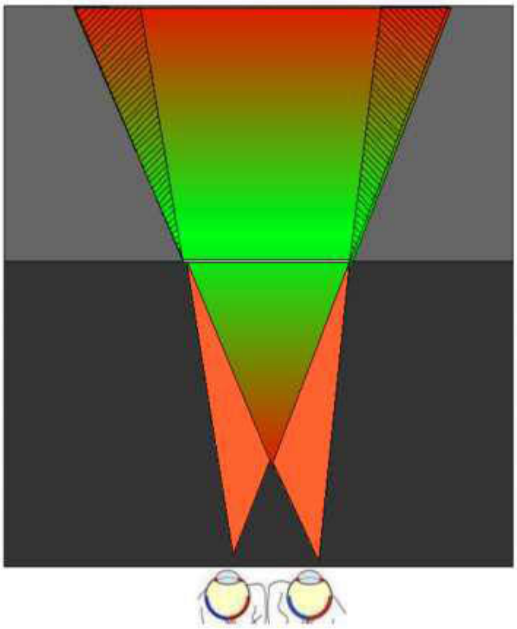
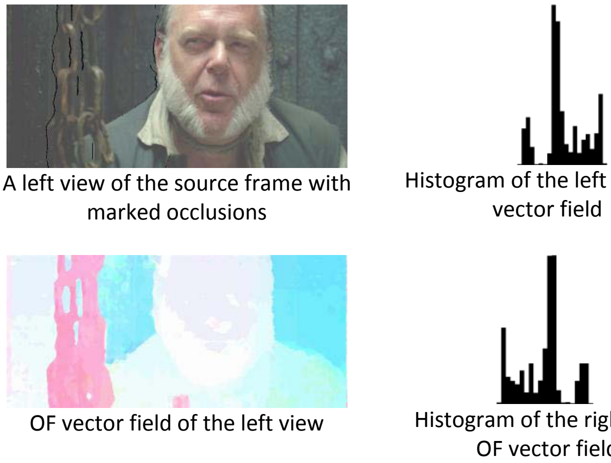


Figure 3. Stereo perception zones. Zone of stereo comfort perception is marked with green color. You can see that negative parallax regions are one-third of the zone of stereo comfort perception, and positive parallax regions are two-third of the zone of stereo comfort perception.

So if depth of foreground objects isn't much different from background depth we can expect that left view will have more positive disparity values than negative and right view will have more negative disparity values than positive. So in such cases a mass center of the left view disparity histogram must have positive coordinate and a mass center of the right view disparity histogram must have negative coordinate (see Figure 4).



A left view of the source frame with marked occlusions

Histogram of the left view OF vector field

OF vector field of the left view

Histogram of the right view OF vector field

Figure 4. An example of OF vector field histograms. You can see that histograms are almost symmetric. Here we have thin occlusions and the mass center of left OF vector field must be rightwards the mass center of right OF right vector field. taken from the movie "Pirates of the Caribbean: On Stranger Tides".

3. THE ALGORITHM

3.1 Steps of the algorithm

1. Preprocessing step: for each view estimate a necessary data :
   - Estimate occlusions using left-right consistency thresholding.
   - Estimate image edges using Canny edge detector[1].
   - Estimate image L*a*b* gradient using Sobel filter[3].
   - Estimate a disparity histogram.
2. For each view for each occlusions side (left or right) calculate a sum of products of:
   - occlusion width,
   - border confidence,
   - inverted distance between the occlusion and the boundary of the each occlusion row.

   We will call these sums *LS* (*left sum*) and *RS* (*right sum*) respectively. So, we obtain four numbers: *LS* and *RS* of the left view (*left LS*, *left RS*) and *LS* and *RS* of the right view (*right LS*, *right RS*).
3. Calculate a confidence sum: *confidence sum = left LS + left RS + right LS + right RS*, compare it with the *confidence threshold*. If it is less than threshold compare moments of disparity histograms. If it is less than the *distance threshold* then views are swapped, otherwise they are not swapped.

   If it is higher than threshold calculate a probability that views of the current frames are not swapped as (*left LS + right RS*) / (*left LS + left RS + right LS + right RS*). If it is less than *probability threshold* then views are swapped, otherwise they are not swapped.

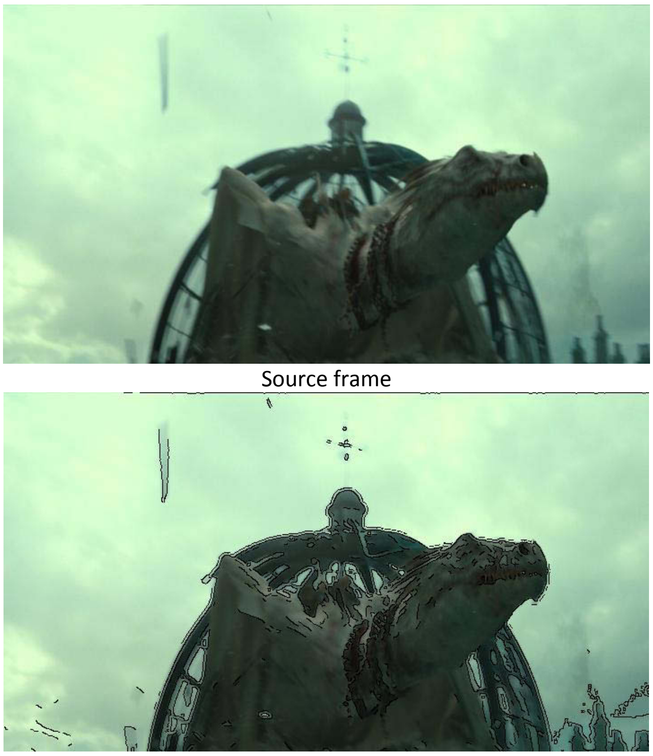
3.2 The detailed algoritm explanation

1. **Occlusions estimation**

. We use LRC thresholding and then perform median filtering of the obtained binary mask.

2. **Edge detection**

We tried simple and reliable Canny algorithm and it produced satisfactory results (see Figure 5).We have chosen parameters for Canny using 30 test sequences from 5 films. In future we will probably use more complicated versions of edge detector.



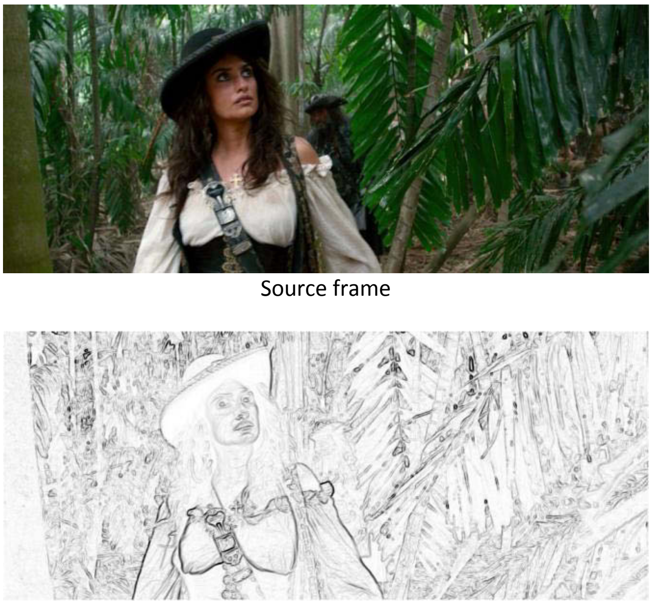
Source frame

Source frame with edges

Figure 5. A source image and an image with marked edges produced by Canny detector. The frame is taken from the movie Harry Potter and the Deathly Hallows.

3. Gradient calculation

We use gradient values as a confidence measure for edges. An example of gradient calculated with Sobel filter is presented in Figure 6.



Source frame

Sobel gradient

Figure 6. An example of Sobel gradient. The source frame is taken from the movie "Pirates of the Caribbean: On Stranger Tides".

4. Disparity histogram

In Figure 4 examples of left and right disparity histograms were presented. You can see that here we have thin occlusions and the mass center of left OF vector field must be rightwards the mass center of right OF right vector field.

5. Occlusion-based confidence values calculation

In this step we calculate how well occlusions boundaries are aligned with image edges. We use the next considerations:

- wide occlusions are more confident than thin occlusions
- edges with higher values of gradient values are more confident
- the closer an occlusion and an edge are – the more likely the correspond to one object's border.

6. Decision making

Calculate a confidence sum: *confidence sum = left LS + left RS + right LS + right RS* (these values were described above), compare it with the *confidence threshold*. If it is less than threshold compare moments of disparity histograms. If it is less than the *distance threshold* then views are swapped, otherwise they are not swapped.

If it is higher than threshold calculate a probability that views of the current frames are not swapped as (*left LS* + *right RS*) / (*left LS* + *left RS* + *right LS* + *right RS*). If it is less than *probability threshold* then views are swapped, otherwise they are not swapped.

So we have two parameters which can be used for algorithm tuning and recall\precision regulating.

4. RESULTS AND ANALYSIS

4.1 Results

For our test set we took 780 random frames from 13 movies, 60 frames from the each movie. These frames were manually checked. For our task it is easy to obtain samples with swapped channels, we need only to swap views of good frames. So we obtain a test set of 1560 frames – 780 frames with non-swapped views and 780 frames with swapped views.

We searched for optimal parameter values in order to minimize false negatives, because our main goal is to find frames with swapped channels in real films. So in our tests we preferred to preserve high recall values and in some cases loose in precision.

Results of our tests are presented as recall/precision diagrams.

We run our algorithm on the test set varying *probability threshold* from 0.01 to 1.0 with all other parameters fixed and obtained the recall/precision diagram, which is presented in Figure 7.

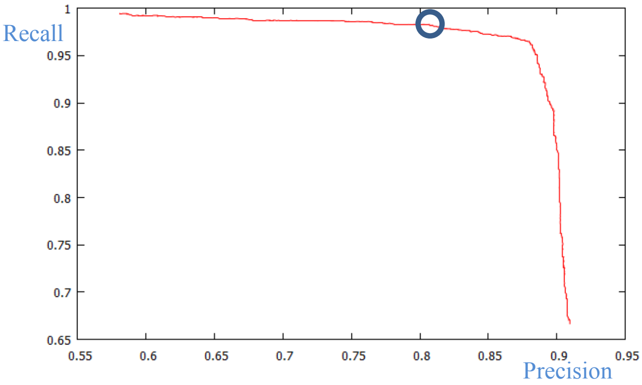


Figure 7.The recall/precision diagram obtained varying *probability threshold* from 0.01 to 1.0 with all other parameters fixed. The test set included 780 frames from 13 movies.

.

We run our algorithm on the test set varying *distance threshold* from -2.0 to 12.0 with all parameters fixed and obtained the next recall/precision diagram Figure 8.

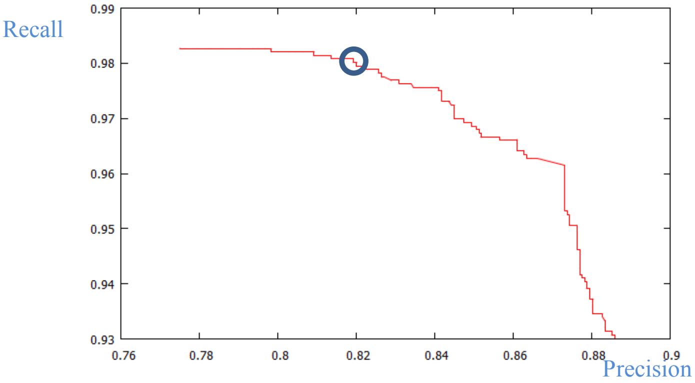
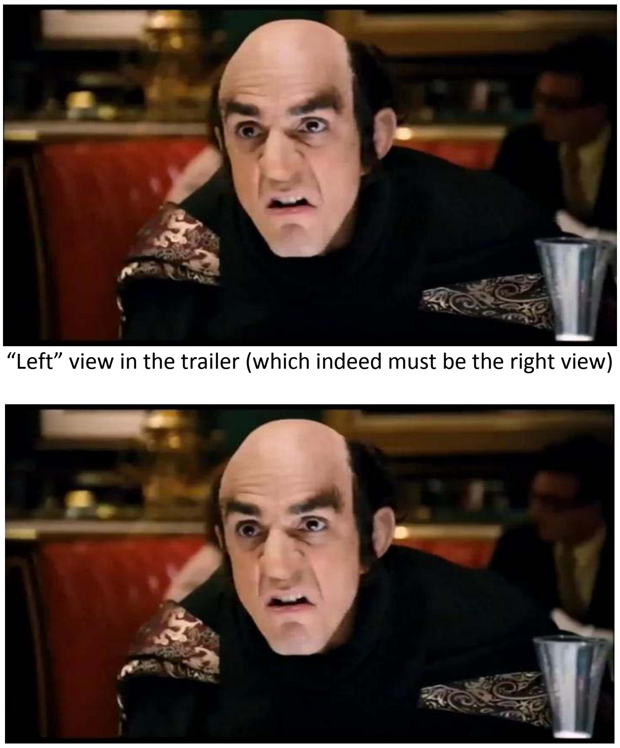


Figure 8.The recall/precision diagram obtained varying *distance threshold* from -2 to 12 with all other parameters fixed. The test set included 780 frames from 13 movies.

Using our algorithm we have found swapped views in the "The Smurfs" trailer. It's frame number 1969 (see Figure 9).



"Left" view in the trailer (which indeed must be the right view)

"Right" view in the trailer (which indeed must be the left view)

Figure 9. Here is the frame with swapped views. It's frame number 1969 form the trailer "The Smurfs".

4.2 Perfomance speed

Algorithm was tested on Intel Corei7-2630QM CPU @ 2.00GHz, 8 cores, 8 GB RAM. The test set contained 780 frames from 13 films.

An average time of work for one frame:

SD (720x480) resolution: 13.79 sec. per frame.

HD (1280x720) resolution: 51.55 sec. per frame

Now regions overlapping occlusions algorithm is based on Optical Flow algorithm implemented on CPU, so most of the time takes Optical Flow computation.

The most time consuming part during the tests was Optical Flow computation, but our optical flow also has GPU implementation and with adapting algorithm to it we will increase speed of processing. We expect speed increase near 5–10 times.

If we apply this algorithm to the whole film, we take only one frame from a scene, because if views are swaped, they are swapped for the whole scene.

It would take about 8 hours to process all scenes from the whole film in SD resolution.

4.3 Further improvements

1. We can combine methods for decision making in a more complicated way, than simple thresholding.
2. We can use the consideration that a color of pixels in occlusion must be similar with background color, not with the object color.
3. There is the fact that on a left view the object with the lowest depth value must have the highest disparity value (if we consider disparity as a signed value, not only its module) and all the way round on a right view – the object with the lowest depth value must have the lowest disparity value (see Figure 2). If we detect where an object is and where a background is, i.e. find out what is further and what is closer, we can make a decision based on this knowledge.
   We can use rough occlusions produced by Motion Estimation for the consequent frames (not between views) and their color similarity with background for local background/foreground segmentation. Also we can use segmentation from motion for objects detection.

5. CONCLUSION

An algorithm for automatic swapped views detection is proposed. It is based on occlusion detection and motion vectors histogram. The algorithm was tested on 780 frames from 13 movies. The recall\precision diagrams were constructed using two parameters. The complexity is estimated. The drawbacks are analyzed and further directions are proposed.

6. AKNOLEDGMENTS

This work is partially supported by the Intel/Cisco Video-Aware Wireless Network (VAWN) Program and by grant 10-01-00697a from the Russian Foundation of Basic Research.

7. REFERENCES

[1] Canny, J., *A Computational Approach To Edge Detection*, IEEE Trans. Pattern Analysis and Machine Intelligence, 8(6):679–698, 1986.

[2] Egnal, G., Wildes, R., P., *Detecting Binocular Half-Occlusions: Empirical Comparisons of Five Approaches,* IEEE Trans. Pattern Analysis and Machine Intelligence, 24(8):1127–1133, 2002.

[3] H. Farid and E. P. Simoncelli, *Differentiation of discrete multi-dimensional signals*, IEEE Trans Image Processing, 13(4):496–508, 2004.

[4] Ogale, S., A., Aloimonos, Y., *Shape and the Stereo Correspondence Problems,* International Journal of Computer Vision, 65(3):147–162, 2005.