# ACCURATE METHOD OF TEMPORAL-SHIFT ESTIMATION FOR 3D VIDEO

**Conference Paper** · June 2018

DOI: 10.1109/3DTV.2018.8478431

# ACCURATE METHOD OF TEMPORAL-SHIFT ESTIMATION FOR 3D VIDEO

*Aleksandr Ploshkin and Dmitriy Vatolin*

Lomonosov Moscow State University, Russia

## ABSTRACT

Video synchronization is a fundamental computer-vision task that is necessary for a wide range of applications. A 3D video involves two streams, which show the scene from different angles concurrently, but many cases exhibit desynchronization between them. This paper investigates the problem of synchronizing the left and right stereoscopic views. We assume the temporal shift (time difference) and geometric distortion between the two streams are constant throughout each scene. We propose a temporal-shift estimation method with subframe accuracy based on a block-matching algorithm.

***Index Terms*** **—** 3D video, quality assessment, temporal shift, spatio-temporal alignment, subframe accuracy.

## 1. INTRODUCTION

Movies in stereoscopic format are popular around the world, but their popularity has declined over the last few years. In the meantime, the number of movies shot in stereo format has dropped every year; filmmakers have instead mainly used computer-graphics techniques or conversion from 2D format. Many viewers experience various kinds of discomfort while watching 3D movies — headaches, eye fatigue, and so on — potentially causing them to lose interest in this format. The situation may owe to several factors, including: imperfect video-transmission technologies and 3D-scene characteristics, such as motion speed, scene volume, brightness, and contrast. The most important factor is the quality of the stereoscopic video.

Many artifacts can cause discomfort and a deteriorating visual experience for viewers watching stereoscopic movies in the theater. A particularly painful example is temporal shift between views in a stereo pair. This shift is the time difference between two video streams, that is: the actions in one video stream are ahead of the same actions in the other video stream. It means the viewer will see situations that are impossible in the real world.

Temporal shift occurs when the two input sequences have a time offset between them (e.g., if the cameras were not activated simultaneously — see the example in Figure 1) and/or when they have slightly different frame rates for some technical reason.

A temporal shift most frequently owes to low-quality shooting equipment. For example, it appears when using a cheap recording system with independent cameras that launch asynchronously. In this case, the temporal shift will be constant throughout the entire scene. Hence, it is necessary to estimate temporal shift with subframe accuracy, meaning the estimated value can be non-integral. Even if recording begins simultaneously, however, a changing time shift can occur if the two cameras have different frame-recording rates.

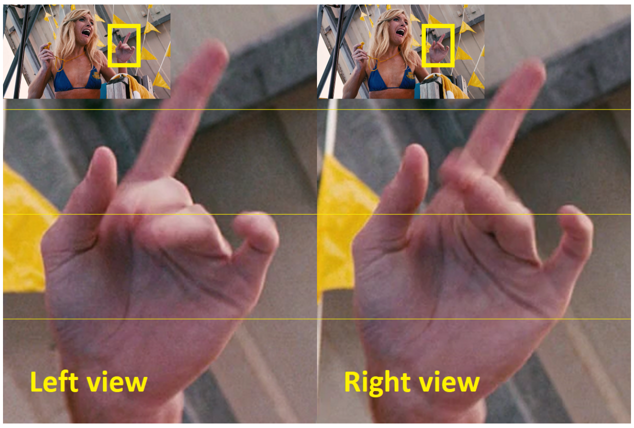


Figure 1: Left and right views from a scene exhibiting temporal shift in *Piranha 3DD* (2012).

## 2. RELATED WORK

The task of detecting distortions that occur during compression and subsequent transmission of a regular video has been well studied, so most research into evaluation of stereoscopic-video quality aims to detect similar artifacts [1, 2]. Relatively few works define and evaluate distortions that arise during production of stereoscopic videos.

Temporal-shift detection can employ methods that perform spatio-temporal alignment of regular video sequences on the basis of trajectory analysis [3, 4, 5, 6], but they have a high computational complexity. Some methods allow temporal synchronization of video from several cameras that shoot the same scene from different angles [7, 8], but they do not assume subframe accuracy. In [9] authors propose a temporal shift estimation algorithm between stereoscopic views with subframe accuracy, however this algorithm strongly depends on geometric distortions between stereoscopic views.

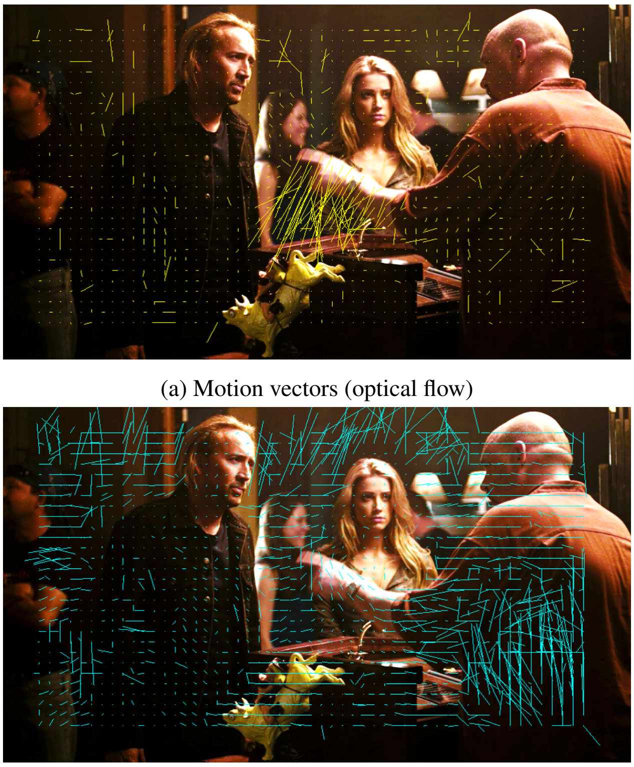
The authors of [10] concluded that temporal shift between stereo views has negative effect on the viewer's experience. They also presented an algorithm for estimating temporal shift, but it is limited to just integer values.

## 3. PROPOSED METHOD

Each video consists of temporal "shots," each of which represents a unique and continuous sequence of frames captured by a camera. Since the problem of scene-boundary detection is well studied, we assume that stereoscopic video is such shot.

We additionally assume that the spatial transformation between the stereoscopic views is constant throughout the scene and that the scene has a constant temporal shift. This assumption corresponds to constant relative camera positions during recording.

978-1-5386-6125-3/18/$31.00 ©2018 IEEE



(a) Motion vectors (optical flow)

(b) Disparity map

Figure 2: Motion and disparity maps obtained using the block-matching algorithm.

In general, the spatial transformation between frames is projective, but because the synchronized videos are stereoscopic views, a 2D affine-transformation model can approximate geometric distortions between the views.

### 3.1. Mathematical formulation

To approximate geometric distortions between stereoscopic views, we chose a 2D affine-transformation model in which we assume each frame in the sequence from one stereoscopic view is rotated and scaled relative to the other view in the same manner throughout the shot. Let $\mathbf{r_L} = [x_L, y_L, 1]^T$ and $\mathbf{r_R} = [x_R, y_R, 1]^T$ be the position vectors (in homogeneous coordinates) of the corresponding points in the left and right stereoscopic views, respectively. In addition, let $\mathbf{v_R} = [v_{x_R}, v_{y_R}, 1]^T$ be the motion vector of this point in the right view (also in homogeneous coordinates) and let $\Delta t$ be the temporal shift between views. We can then construct an equation that describes the spatio-temporal transformation:

$$\mathbf{r_L} = M \times \mathbf{r_R} + \Delta t \times \mathbf{v_R}. \tag{1}$$

Here, $M \in \mathbb{R}^{3\times 3}$ is the following affine transformation matrix:

$$M = \begin{bmatrix} \beta+1 & \alpha & \delta_x \\ -\alpha & \beta+1 & \delta_y \\ 0 & 0 & 1 \end{bmatrix}, \tag{2}$$

where $[\delta_x, \delta_y, 1]^T$ is the translation vector. We define the parameters $\alpha = k \sin\varphi$ and $\beta = k\cos\varphi - 1$, where $k$ is the scale coefficient, and $\varphi$ is the rotation angle. Let $\mathbf{d} = \mathbf{r_L} - \mathbf{r_R} = [d_x, d_y, 0]^T$ be the disparity vector of the corresponding point; we can then represent (1) as:

$$\mathbf{d} = (M - I) \times \mathbf{r_R} + \Delta t \times \mathbf{v_R}, \tag{3}$$

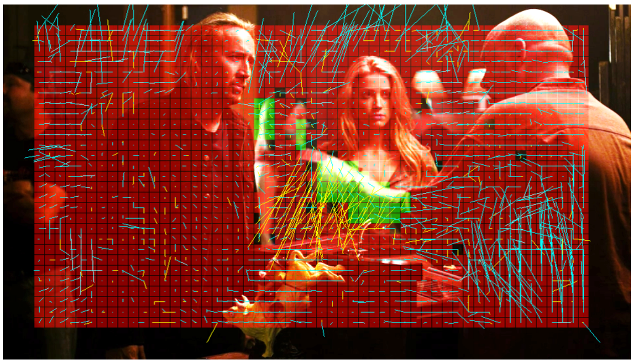


Figure 3: Block extraction. Green indicates blocks with sufficient confidence, red indicates blocks with insufficient confidence. Our method excludes all red blocks.

where $I$ denotes the $3 \times 3$ identity matrix.
Stereoscopic video can employ horizontal disparity by design to achieve the stereo effect, but vertical disparity is always the result of spatio-temporal misalignment. Therefore, we extract the vertical component from (3):

$$d_y = -\alpha \cdot x_R + \beta \cdot y_R + \Delta t \cdot v_{y_R} + \delta_y \tag{4}$$

Therefore, we must solve the regression problem for the parameters $\alpha$, $\beta$, $\Delta t$, and $\delta_y$.

### 3.2. Temporal-shift estimation

To obtain the disparity map $\mathbf{d}$ and optical flow $\mathbf{v}$ for each frame, we use a block-matching algorithm [11], which calculates the vertical and horizontal projections of the disparity and motion vectors for each block of $n \times n$ pixels (in our case, $n = 16$). Figure 2 illustrates the motion and disparity vectors we obtain from this algorithm. Since both the disparity map and optical flow can be noisy, we construct the corresponding confidence maps on the basis of block intensity variance (uniform areas are penalized) and the left-right checking (LRC) criterion [12]. To reduce the impact of noise, we exclude blocks for which we calculated a low confidence as well as blocks that have a $\mathbf{v_R}$ value of approximately 0, as Figure 3 shows. We choose the coordinates $x_R$ and $y_R$ in (4) as center of the corresponding block.
We accumulate data throughout the shot, obtaining numerous equations, such as (4), for different blocks and different frames:

$$\begin{bmatrix} -x_R^{(1)} & y_R^{(1)} & v_{y_R}^{(1)} & 1 \\ \vdots & \vdots & \vdots & \vdots \\ -x_R^{(m)} & y_R^{(m)} & v_{y_R}^{(m)} & 1 \end{bmatrix} \times \begin{bmatrix} \alpha \\ \beta \\ \Delta t \\ \delta_y \end{bmatrix} = \begin{bmatrix} d_y^{(1)} \\ \vdots \\ d_y^{(m)} \end{bmatrix}, \tag{5}$$

where $m$ is the number of blocks with sufficient confidence.
Finally, to approximate system (5) and to determine the temporal shift as well as the affine-transformation parameters from (4), we use the RANSAC algorithm [13], which is robust to noisy data.
The following is a simplified scheme of our proposed algorithm:

Figure 4: Left and right views of the test shot, in which the right view is rotated by 0.72 degrees, scaled by 1.8%. The temporal shift between views is 0.5 frames.

**Algorithm 1** Temporal-shift estimation

```
procedure COMPUTETEMPORALSHIFT($S$)    ▷ $S$ is the shot in the stereoscopic video
    $A \leftarrow \varnothing$    ▷ $A$ is the left-hand side of (5)
    $b \leftarrow \varnothing$    ▷ $b$ is the right-hand side of (5)
    for $F_L, F_R \in S$ do    ▷ $F_L$ and $F_R$ are the left and right frames, respectively
        $D_y \leftarrow$ BlockMatching$(F_L, F_R)$
        $V_y \leftarrow$ BlockMatching$(F_R^{prev}, F_R)$
        for $\langle d_y, v_y \rangle \in \langle D_y, V_y \rangle$ do
            if block is good then
                $A \leftarrow A \cup \langle x, y, v_y, 1 \rangle$
                $b \leftarrow b \cup \langle d_y \rangle$
            end if
        end for
    end for
    $\langle \alpha, \beta, \Delta t, \delta_y \rangle \leftarrow$ RANSAC$(A, b)$
    return $\Delta t$
end procedure
```

## 4. EXPERIMENTAL RESULTS

To test our proposed algorithm, we created a set of 396 scenes from the 3D version of *Titanic*. We chose this film because the stereoscopic version was produced using 2D-to-3D conversion, precluding noninteger temporal shifts. We verified that all scenes in the test set have zero temporal shift by making comparison between left and right view. Out of all 396 scenes, 99 have no temporal shift or geometric distortions, 99 have only temporal shift by one value (0.25, 0.5, 1.0, or 2.0), and 99 have only geometric distortions modeled by applying an affine transformation to one view using the following parameters:

- the center is a random point in the frame
- the angle has the distribution $N(0, 0.4^2)$
- the scaling coefficient $k = 1 + |\xi|$ (only zoom), where $\xi$ has the distribution $N(0, 0.01^2)$

Here, $N(\mu, \sigma^2)$ is the normal distribution with mean $\mu$ and variance $\sigma^2$. The remaining 99 scenes in our test have both temporal shift and geometric distortions. Figure 4 shows one example.

We implemented the proposed temporal-shift estimation algorithm based on the Equation (4) in C++ using the OpenCV library, then tested it on 2.4GHz Intel Core i7-4700HQ processor with 12GB of memory. In provided experiment the number of blocks $m$ in the Equation (5) is defined as 5% of all blocks in a scene and at the same time not less than 5 000 for optimal trade-off between computational speed and quality. The average speed of the temporal-shift computation is 0.9 fps.

We compared the proposed algorithm with the temporal-shift estimation algorithm described in [9]. First of all, we may propose binary classification of scenes. First group includes scenes where the temporal shift is correctly estimated by an algorithm. Second group is consist of scenes with incorrectly estimated temporal shift. We assume that the estimated temporal shift is correct, when the deviation between the output of the algorithm and the ground-truth is less than threshold given. For the quality comparison we used the following measures:

- classification accuracy with fixed threshold of deviation
- the root-mean-square deviation (RMSD) in the entire test set

Accuracy is the proportion of true results (both true positives and true negatives) among the total number of cases examined for the given threshold, in which the result is assumed correct. In the case described it is the measure of method's degree of precision of the temporal-shift estimation. We choose the threshold $T = 0.1$ frames, because smaller shifts are invisible for human. Figure 5 shows the accuracy for different values of threshold for the test results. The higher line on the graph, the more precise temporal-shift estimation. For the threshold $T = 0.1$ frames accuracy of the proposed algorithm is approximately 0.98 while accuracy of [9] is less than 0.82. The RMSD of our algorithm is less than 0.001 frames whereas algorithm from [9] gives more than 0.25 frames.

Next, we analyzed 106 feature stereoscopic movies using our algorithm and detected 515 shots exhibiting temporal shift; Table 1 provides further details. Most of these shots had a temporal shift of less than 0.1 frames, but a few had a large shift of up to 2 frames.

| | |
|---|---|
| Movies analyzed | 106 |
| Shots with temporal shift | 515 |
| Movies with temporal shift | 26 |
| Movies with temporal shift (percentage) | 25.47% |
| Average temporal shift | 0.21 frames |
| Maximum temporal shift | 2 frames |
| Average duration | 7.78 s |
| Total duration | 1h 6m |

Table 1: Movie analysis.

## 5. CONCLUSION

In this paper, we propose a temporal-shift estimation algorithm with subframe accuracy that is robust to geometric distortions between stereoscopic views. Testing of our implementation on a set of 396 scenes with and without temporal shift and geometric distortions revealed high accuracy. Our algorithm is more accurate than the algorithm from [9]. In addition, we used our algorithm to analyze 106 stereoscopic movies and found more than 500 scenes in 26 of them that contain temporal shift.

## 6. REFERENCES

[1] Atanas Boev, Danilo Hollosi, Atanas Gotchev, and Karen Egiazarian, “Classification and simulation of stereoscopic artifacts in mobile 3DTV content,” in *Stereoscopic Displays*

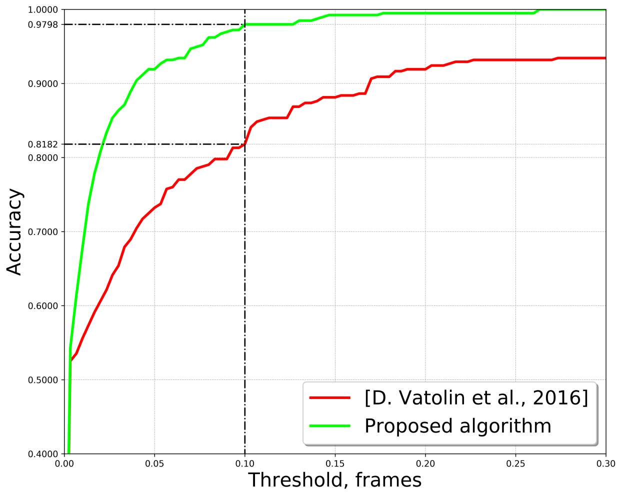


Figure 5: Accuracy graph for the test set. Accuracy for the given threshold measured in frames, in which the result is assumed correct.

*and Applications XX*. International Society for Optics and Photonics, 2009, vol. 7237, p. 72371F.

[2] Quan Huynh-Thu, Patrick Le Callet, and Marcus Barkowsky, "Video quality assessment: from 2D to 3D challenges and future trends," in *Image Processing (ICIP), 2010 17th IEEE International Conference on*. IEEE, 2010, pp. 4025–4028.

[3] Yaron Caspi and Michal Irani, "Spatio-temporal alignment of sequences," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 11, pp. 1409–1424, 2002.

[4] Benjamin Meyer, Timo Stich, Marcus A. Magnor, and Marc Pollefeys, "Subframe temporal alignment of non-stationary cameras," in *BMVC*, 2008, pp. 1–10.

[5] Georgios D. Evangelidis and Christian Bauckhage, "Efficient and robust alignment of unsynchronized video sequences," in *Joint Pattern Recognition Symposium*. Springer, 2011, pp. 286–295.

[6] Georgios D. Evangelidis and Christian Bauckhage, "Efficient subframe video alignment using short descriptors," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 10, pp. 2371–2386, 2013.

[7] Oliver Wang, Christopher Schroers, Henning Zimmer, Markus Gross, and Alexander Sorkine-Hornung, "Videosnapping: interactive synchronization of multiple videos," *ACM Transactions on Graphics (TOG)*, vol. 33, no. 4, pp. 77, 2014.

[8] Matthijs Douze, Jérôme Revaud, Jakob Verbeek, Hervé Jégou, and Cordelia Schmid, "Circulant temporal encoding for video retrieval and temporal alignment," *International Journal of Computer Vision*, vol. 119, no. 3, pp. 291–306, 2016.

[9] Dmitriy Vatolin, Alexander Bokov, Mikhail Erofeev, and Vyacheslav Napadovsky, "Trends in S3D-Movie Quality Evaluated on 105 Films Using 10 Metrics," *Electronic Imaging*, vol. 2016, no. 5, pp. 1–10, 2016.

[10] Lutz Goldmann, Jong-Seok Lee, and Touradj Ebrahimi, "Temporal synchronization in stereoscopic video: influence on quality of experience and automatic asynchrony detection," in *Image Processing (ICIP), 2010 17th IEEE International Conference on*. IEEE, 2010, pp. 3241–3244.

[11] Aroh Barjatya, "Block matching algorithms for motion estimation," *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 3, pp. 225–239, 2004.

[12] Geoffrey Egnal and Richard P. Wildes, "Detecting binocular half-occlusions: empirical comparisons of five approaches," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 8, pp. 1127–1133, 2002.

[13] Martin A. Fischler and Robert C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," in *Readings in Computer Vision*, pp. 726–740. Elsevier, 1987.