

*Regular Article*

# An Extended Occlusion Detection Approach for Video Processing

Synh Viet-Uyen Ha, Tuan-Anh Vu, Ha Manh Tran

School of Computer Science and Engineering, International University, Vietnam National University, Ho Chi Minh City, Vietnam

Correspondence: Synh Viet-Uyen Ha, hvusynh@hcmiu.edu.vn

Communication: received 1 March 2018, revised 23 May 2018, accepted 8 August 2018

Online publication: 27 October 2018, Digital Object Identifier: 10.21553/rev-jec.198

The associate editor coordinating the review of this article and recommending it for publication was Dr. Le Hoai Bac.

**Abstract**– Occlusions become conspicuous as failure regions in video processing when unified over time because the contraventions of the restriction of brightness have accumulated and evolved in occluded regions. The accuracy at the boundaries of the moving objects is one of the challenging areas that required further exploration and research. This paper presents the work in process approach that can detect occlusion regions by using pixel-wise coherence, segment-wise confidence and interpolation technique. Our method can get the same result as usual methods by solving only one Partial Differential Equations (PDE) problem; it is superior to existing methods because it is faster and provides better coverage rates for occlusion regions than variation techniques when tested against a varied number of benchmark datasets. With these improved results, we can apply and extend our approach to a wider range of applications in computer vision, such as background subtraction, tracking, 3D reconstruction, video surveillance, video compression.

**Keywords**– optical flow, unstable region, object boundaries, occlusion detection, video object extraction, video object segmentation.

## 1 INTRODUCTION

Occlusions is an important feature in the field of computer vision because they present information of occluded region in two continuous frames of the scene. Occlusions occurs because no 2-D image can adequately capture the content of a 3-D image; therefore, when we pair the contents of two images by using optical flow, occlusion regions are not a redundant but an information source of the geometry of the scene [1]. However, occlusions detection due to a research topic is omitted as it requires estimation of the scene and movement of the camera, both of which are problems of their own. As a result, current techniques for detecting occlusions occur in two steps: first they estimate the relative motion of the camera and the scene with the optical flow, and then detect these occlusions as regions where optical models fail. By breaking down the problem into two, these techniques can be built by linking existing algorithms but not modeling the process that causes occlusions.

There are two main approaches according to the order of using motion information in moving object segmentation. The first approach uses motion information as the primary segmentation criterion. The position and shape of the moving object is detected from the frame difference of two consecutive frames. Then object boundaries are determined from spatial or temporal information. This approach runs so fast, but it gets a low accuracy result if objects move too fast or complex.

In the second approach, spatial homogeneity instead of motion information as the primary segmentation criterion. This approach consists of two steps: first, a

segmentation algorithm is used to find homogenous regions; after that, the motion vector of each region is calculated and the regions with similar motion are merged together. The object boundary results of this approach are more precise than the results of the above approach. It does not take into account stationary background and it requests only two consecutive frames. However, this approach has to solve two PDE problems (one for segmentation algorithm, another one for motion estimation algorithm), so it runs slower than the first approach.

In our proposed method, we use motion information as the primary segmentation criterion, but we use motion estimation instead of change detection. Figure 1 presents an overview of the proposed method.

Comparing with previous methods, our method has following advantages:

- 1) **High accuracy at object boundaries.** The second approach makes good shortcomings of the first one such as background request stationary, objects move too fast, objects have holes, etc... because it uses a segmentation algorithm to find homogenous regions first. However, segmentation is a dilemma because small segments may not contain enough data, and segmentation at a large scale is unreliable. In the proposed method, we classify motion vectors in occlusion regions to find object boundaries. This helps us get a better result.
- 2) **Fast processing speed.** The first approach has the highest processing speed, but it has some shortcomings presented in the above paragraph. The second approach gets a high accuracy result,

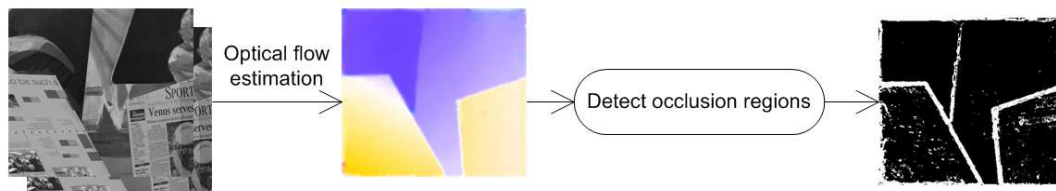


Figure 1. Overview of the proposed method.

but the computational complexity is quite high. Our method is slower than the first approach but it is faster than the second one. Instead of solving two PDE problems, we solve only one.

In this paper, we present a method that can figure out the moving objects boundaries by calculating only one partial differential equation problem. It means that the estimation of optical flow procedure is only called once and this does not use boundary detection method or the segmentation method. We also use interpolation to eliminate aliasing along the space to increase accuracy since the blur filter can damage the edges in the original image and the result.

The rest of this paper is organized as follows: Section 2 reviews related methods. Section 3 introduces the proposed method. Section 4 presents the experimental results and discussion to illustrate the application and usefulness of the proposed algorithm. Section 5 concludes this study with a discussion.

## 2 LITERATURE REVIEW

The variant methods [2, 3] are advanced optical-flow because they provide specific density estimates and lead to minimization problems that can be resolved efficiently [4, 5]. However, two significant issues remain open: the stability of the method for large displacement, and the calculation of optical-flow in occlusion areas.

It is difficult to detect large displacement because the brightness restriction of the optical-flow models cannot be adjusted, the maximum minimization problem is not convex, and numerical algorithms converge to a local minimum point close to the initial point. In order to mitigate the impact of the initialization, classical techniques utilize a multidisciplinary strategy to find the minimum model at large-scale but neglect the small-scale structures that are not in the raw resolution. Large displacement methods [3, 6, 7] solve this problem by introducing a descriptive matching step with the variation model that pilots the multi-resolution to a local minimum relevant for small-scale structures. The various methods are consequently still the core of the optical flow, but only some of which include obvious occlusions in the model.

Flow errors occur in occlusion areas because the restriction of brightness required to incorporate intensity in areas that have no correspondence. To avoid these errors, occlusion-aware techniques are taking into account occlusions in the model. This can be done completely by ignoring the restriction of brightness in

areas where the flow model is broken [8–10] or clearly by presenting an occlusion variable in the [11–15]. A second criterion distinguishing between these methods is the way that occlusions are combined into the model: at first computing the flow of neglected occlusions, using weak flow to detect occlusions, and then adjust the flow in the occlusion areas [11, 14, 15]; while the joint methods [12, 13] clearly present the occlusions in the model and produce a single reduction that the flows and occlusion variables interact. It is more challenge to optimize the joint methods, but these models are stronger because the flow and occlusions both illustrate the data. Therefore, we propose a generic model but design it to indicate occlusion detection rather than the estimation of flow.

There are two criteria for detecting occlusions from estimated flow that can be individual of the level of collaboration between flow and occlusions: the first one using the flow to detect occlusions as unrecognized pixels [9, 12], and the second use non-correspondence between forward and backward flows to detected occlusions as pixels [8, 10, 16]. We accept the first criterion and present a transient model for the set of occlusion; this distinguishes our model from the current technique of processing only two images and ignore the temporary dimension of occlusions in the video.

Although relevant, methods for detecting the boundaries of occlusion regions [17–22] and layered models [17, 19, 23–27] explain a different problem. They detect boundaries of occlusion regions from one segment of the image to find objects in the scene and their relative order; as a result, they are closer to image and motion segmentation than to our method. Recently, machine-learning classifiers have also been used for occlusion detection in [14, 15]. The nature of learning from the data of these approaches differs from our model, which is designed form the restriction of physical rather than data analysis and does not require training sessions.

In summary, we propose a variant method for estimating optical flow and detecting occlusion regions, but focusing on the occlusions in our modeling design. As a variation model, our approach has the adaptability to combine the large displacement techniques [3, 6, 7], and more robust data terms [28] to improve estimation of optical flow.

## 3 THE PROPOSED METHOD

In this session, we introduce our proposed method to detect occlusion regions in the optical flow estimation result to reduce the regions which need to be processed.

**Algorithm 1:** Detect Occlusion Regions

---

```

1 Algorithm: Detecting Occlusion Function
   Input:
    $I$ : the original image  $I_t$ .
    $u, v$ : the results of optical flow estimation.
   Output:
    $Mask$ : the mask consisting of all unstable pixels
2 begin
   /* Find Surjection Pixels */
3   - initialize buffer  $headImg$  used to find the
   pixels which are overlapped in  $I_{t+1}$ ;
4   foreach pixel  $Y(x_Y, y_Y)$  in  $I$  do
5     - compute the correspondent pixel  $Y_{next}$ 
   of the pixel  $Y$  in  $I_{t+1}$ 
6      $x_{Y_{next}} = \text{int}(x_Y + u_Y + 0.5)$ ;
7      $y_{Y_{next}} = \text{int}(y_Y + v_Y + 0.5)$ ;
8     if  $Y_{next}$  is out of the image  $I_{t+1}$  then
9       |  $M_Y = 1$ ;
10    else
11      if the position  $Y_{Next}$  in  $headImg$  was
   marked then
12        |  $Y_{conflict} = headImg_{Y_{next}}$ ;
13        |  $M_{Y_{conflict}} = 1$ ;
14        |  $M_Y = 1$ ;
15      else
16        |  $headImg_{Y_{next}} = Y$ ;
   /* Find Injection Pixels */
17   - initialize buffer  $tailImg$  used to find the
   pixels which are undetermined in  $I_{t+1}$ ;
18    $tailImg = headImg$ ;
19   - use Erode filter on  $tailImg$  to find the pixels
   which are neighbor of undetermined pixels;
20   foreach pixel  $X$  in  $tailImg$  do
21     if  $tailImg_X$  is unmarked AND  $headImg_X$  is
   marked then
22       |  $Y_{undetermined} = headImg_X$ ;
23       |  $M_{Y_{undetermined}} = 1$ ;
24   - restore good features in  $M$ ;
25   - use Eq. 1 to assess stableness of the data
   constraint;
26   - use Eq. 2 to estimate stableness of updating
   process;
27   - use Eq. 5, Eq. 8 and Eq. 9 to find unstable
   regions  $Mask$ ;

```

---

We presented how to detect occlusion regions in Algorithm 1. In the Algorithm 1, to illustrate the capability of our algorithm, we combined it with the Brox's theory [29]. The Brox's energy function is based on three assumptions: a grey value constancy assumption, a gradient constancy assumption, and a smoothness assumption. When an object move from time  $t$  to time  $(t + 1)$ , it creates two instability areas: Region  $U'$ , at the head of motion vectors and Region  $U''$ , at the tail of motion vectors. There are two groups of pixels creating instability in Region  $U'$ . The first group contains

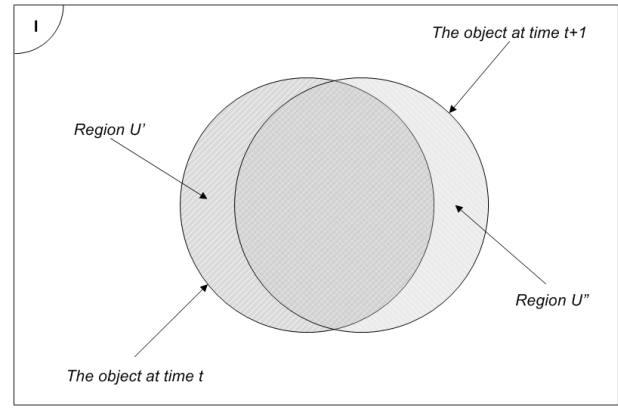


Figure 2. Main areas cause instability of the flow estimation.

pixels moving outside the next frame. Estimation is confused between these pixels moving out the image, and changes in shape of the objects. The second group contains pixels overlapping together in next frame. A part of the background/object (when objects overlap) in the first frame disappears in the next frame, because the objects move or change shape. Estimation around the brightness constancy assumption is incorrect for these pixels. In Region  $U''$ , a part of the background / object absent from the previous frame appears in the next frame. The smoothness assumption among pixels is broken. Thus, their estimation is incorrect. If objects change shape, unstable pixels can result. We do not have enough information for estimating motion vectors in these regions. So, they are unstable regions. In another way, these pixels in these regions will be assessed approximately. As you can see, both Region  $U'$  and Region  $U''$  create the occlusion boundary of the moving object (Figure 2).

We define some concepts used in this algorithm: bijection pixel, surjection pixel, and injection pixel.

**Definition 1** (Bijection pixel). A pixel satisfies three conditions:

- It is in the first frame.
- In optical flow estimation from the first frame to the next frame, it points to a pixel  $x$  in the next frame.
- In optical flow estimation from the next frame to the first frame, the pixel  $x$  points to it (Figure 3).

**Definition 2** (Surjection pixel). A pixel in the first frame points to a pixel which was pointed by at least another pixel. In this case, we have a virtual pixel consisting of all pixels are out of the next frame (Figure 3).

**Definition 3** (Injection pixel). A pixel in the first frame points to a pixel which is adjacent an undetermined pixel. An undetermined pixel in the next frame has no pixel in the first frame pointing to (Figure 3).

**Definition 4** (Stable pixel and unstable pixel). A stable pixel is a bijection pixel. An unstable pixel is a surjection pixel or an injection pixel.

**Definition 5** (Unstable region). A set of unstable pixels which are connected together.

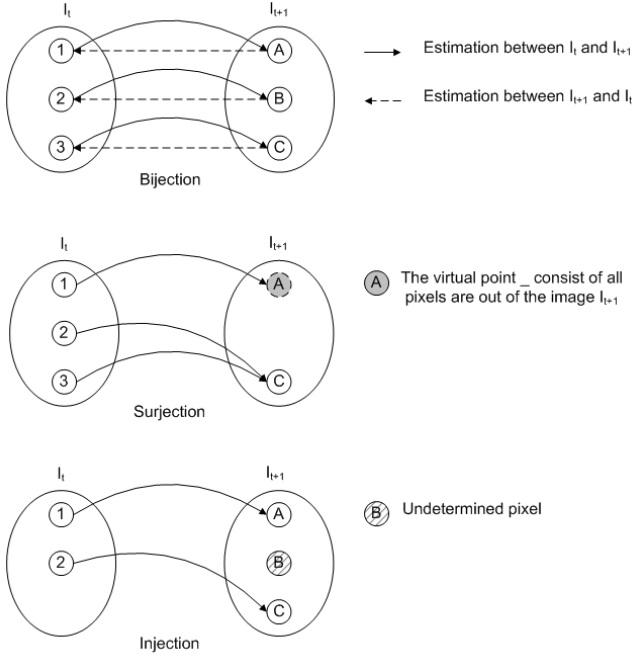


Figure 3. Bijection pixel, surjection pixel and injection pixel.

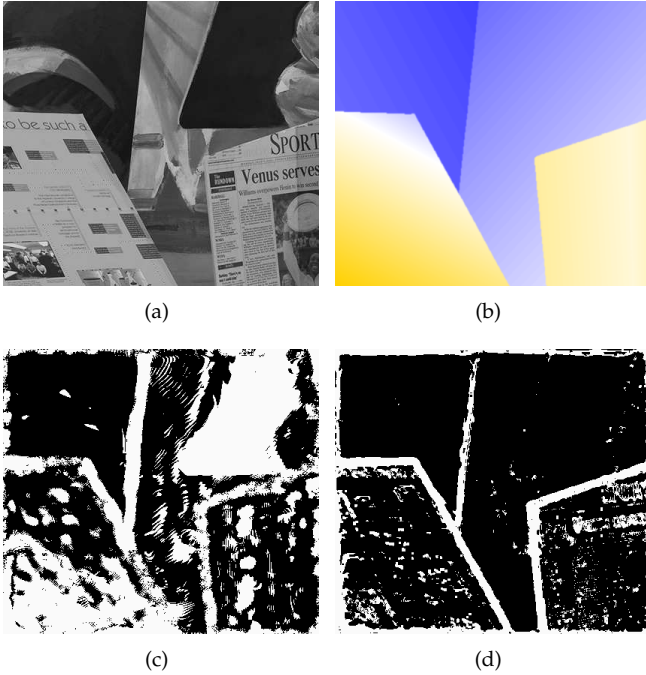


Figure 4. Frame 10 of the Venus sequence. (a) Original image. (b) Groundtruth. (c) Results of determining unstable regions without restricting error of calculation. (d) Results of determining unstable regions with restricting error of calculation.

Assume  $f$  be a function defined on a set of positions  $S$  (a local region in  $I_t$ ) and taking positions in a set  $R$  (a local region of  $I_{t+1}$ ). A pixel  $x$  is a stable pixel if and only if  $f(x)$  is a bijection (one-to-one and onto). In another way,  $x$  is an unstable pixel (it depends on an occlusion region). In our algorithm, we determine unstable pixels directly by finding surjection pixels and injection pixels.

Using the conditions for overlap pixels and undetermined pixels to find unstable regions implies that the values of a motion vector  $(u, v)$  are converted from real

numbers into integers. This creates a calculation error (Figure 4(c)). To restrict the effect of this error, we used good features [30] and extended two ideas presented in [31]. Good features in unstable regions would be restored before these regions could be readjusted.

Regarding pixel-wise coherence  $W_p$ , we only used the data constraint of optical flow estimation. If the difference between the grey value at position  $(i, j)$  in  $I_t$  and the one at position  $(i + u, j + v)$  in  $I_{t+1}$  is large, that pixel may be unstable (Equation (1)).

$$W_p(\mathbf{w}, \mathbf{x}) \triangleq \exp\left(-\frac{|I_{t+1}(\mathbf{x} + \mathbf{w}) - I_t(\mathbf{x})|^2}{\sigma_I^2}\right), \quad (1)$$

where  $\mathbf{w}$  is the motion vector at pixel  $\mathbf{x}$ .

Regarding segment-wise confidence  $W_s$ , we focused on the pixels in the neighbor region of the pixel processing and which have similar color. If the motion vector at a pixel changes sharply, it can be unstable (Equation (2)).

$$W_s(\mathbf{w}^f, \mathbf{w}^0, \mathbf{x}_c) = \frac{\sum_{\mathbf{x} \in Nb} \exp\left(-\left\|\mathbf{w}_x^f - \mathbf{w}_x^0\right\|^2 \frac{Ep(\mathbf{w}_x^0, \mathbf{x})}{\sigma_A^2}\right) \kappa(\mathbf{x}, \mathbf{x}_c)}{\sum_{\mathbf{x} \in Nb} \kappa(\mathbf{x}, \mathbf{x}_c)}, \quad (2)$$

where  $Nb$  is a neighbor region of pixel  $\mathbf{x}_c$ ,  $\mathbf{w}_x^0$  is the motion vector at pixel  $\mathbf{x}$  in the initial step,  $\mathbf{w}_x^f$  is the motion vector at pixel  $\mathbf{x}$  in the final step, and  $\kappa(\mathbf{x}, \mathbf{x}_c)$  constrains that two adjacent pixels must have similar color.

$$\kappa(\mathbf{x}, \mathbf{x}_c) = (1 - M(\mathbf{x})) \exp\left(-\frac{|I_t(\mathbf{x}) - I_t(\mathbf{x}_c)|^2}{\sigma_s^2}\right) \quad (3)$$

$$M(\mathbf{x}) = \begin{cases} 1 & \text{if pixel } \mathbf{x} \text{ is unstable} \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

Then, the unstable regions is computing as below:

$$Com(\mathbf{w}^f, \mathbf{w}^0, \mathbf{x}_c) = \begin{cases} \zeta \cdot W_s(\mathbf{w}^f, \mathbf{w}^0, \mathbf{x}_c), & \text{if } M(\mathbf{x}) = 0 \\ W_p(\mathbf{w}_{x_c}^f, \mathbf{x}_c) \cdot W_s(\mathbf{w}^f, \mathbf{w}^0, \mathbf{x}_c), & \text{otherwise,} \end{cases} \quad (5)$$

where  $\zeta$  is a constant to penalize the unstable pixel.

However, the pixels near the boundary of the unstable regions in Equation (5) will create aliasing along spatial dimensions. Such aliasing effects can cause sudden changes in the computation of feature vector. To avoid this, we use bi-linear interpolation of the pixel weight into the spatial orientation. There are many interpolation method such as: nearest neighbor interpolation, bi-linear interpolation, bi-cubic interpolation and cubic B spline interpolation. But bi-linear interpolation has continuous results, average computational time and good visual effect [32]. Thus, the bi-linear interpolation is chosen to use in our proposed method.

Let  $h(x, y)$  is the value of the interpolation function at pixel with the coordinates  $x$  and  $y$  in  $Com(\mathbf{w}^f, \mathbf{w}^0, \mathbf{x}_c)$  (Figure 5). We calculate linear interpolation in the  $x$ -

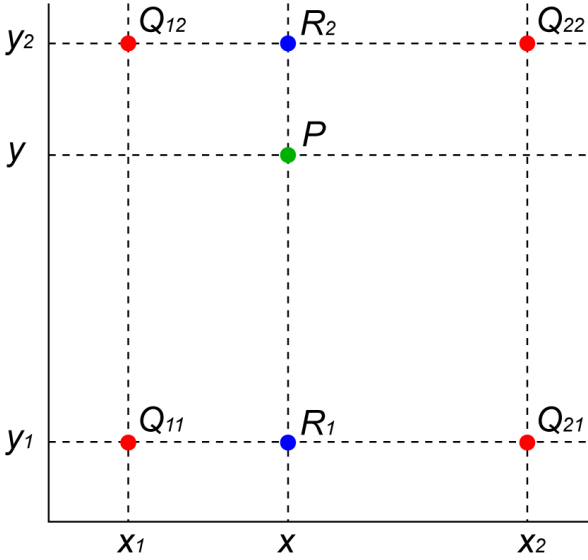


Figure 5. Red dots show the value ant each pixel and the green dot is the interpolated position.

direction as follow:

$$h(x, y_1) \approx \frac{x_2 - x}{x_2 - x_1} h(Q_{11}) + \frac{x - x_1}{x_2 - x_1} h(Q_{21}) \quad (6)$$

$$h(x, y_2) \approx \frac{x_2 - x}{x_2 - x_1} h(Q_{12}) + \frac{x - x_1}{x_2 - x_1} h(Q_{22}) \quad (7)$$

Then, we calculate linear interpolation in the  $y$ -direction to get the final interpolated value:

$$h(x, y) \approx \frac{y_2 - y}{y_2 - y_1} h(x, y_1) + \frac{y - y_1}{y_2 - y_1} h(x, y_2) \quad (8)$$

where  $Q_{11}$ ,  $Q_{12}$ ,  $Q_{21}$ ,  $Q_{22}$  is the pixel with the following coordinates respectively  $(x_1, y_1)$ ,  $(x_1, y_2)$ ,  $(x_2, y_1)$ ,  $(x_2, y_2)$ .

The final unstable regions are computed as:

$$F(\mathbf{w}^f, \mathbf{x}_c) = \begin{cases} 1 & \text{if } h(x, y) \geq \xi \\ 0 & \text{otherwise} \end{cases} \quad (9)$$

where  $\xi$  is a threshold coefficient. And  $\xi = 0.1$  can be practically selected on a validation set.

## 4 EXPERIMENTS AND DISCUSSION

We compare the proposed method to three different variational models: Alvarez *et al.* [8], Ayvaci *et al.* [13] and Estellers *et al.* [33].

To evaluate the proposed algorithm, we performed a number of experiments using the MPI-Sintel public flow dataset [34]. This dataset is a computer-generated database that has ground-truth occlusions. We use the final pass of the data set, including shading, blur and atmospheric shading in the video sequence to cover all the cases. Each sequence has 50 frames with the resolution of  $1024 \times 436$ , with 24 frames per second. We tested five frames in each sequence to produce the best results for MPI-Sintel. Experiments have been conducted to measure the accuracy of the proposed method and to test the system performance as a whole. Qualitative and

quantitative evaluations of the proposed algorithm have been carried out.

### 4.1 Qualitative Evaluation

In this section, we will show our Qualitative evaluation on the five sequences *alley1*, *bamboo1*, *bandage1*, *bandage2* and *market2* of the MPI-Sintel dataset [34].

The figure 9 and 10 present detected occlusion regions for sequences *bandage2* and *market2*. Our method have better results (over 30%) than the runner-up method because our method has the least false alarm rate(TPR). Especially, in the figure 10, our method can detect the region of two women in the right side, but the others can not.

The Figure 6 shows an unsuccessful case in the *alley1* sequence of our model. Several reasons reduce the accuracy of our model. For examples, the combination of the girl's hair oscillations and the non-smooth displacements of the camera that create some noises when detecting occluded regions. Our model also detects occluded regions of object on the left, but the boundaries of the object is a bit thicker than ground-truth. Similarly, the Figure 7 and 8 show the other two of failure cases in the sequence *bamboo1* and *bandage1*. In the Figure 7, our model can detect almost occluded regions, but similar to Figure 6, the bamboo oscillations also create some noises to reduce the accuracy of our model. In the Figure 8, the baby dragon's skin, the girl's gloves and the background are very similar, so these cause false detection of baby dragon's boundaries. However, our results are slightly lower than the best results. For a more detailed analysis of these models, we use a quantitative measure in the next session.

### 4.2 Quantitative Evaluation

To evaluate Quantitative of proposed method, we use the theory by [35–38] to calculate precision or positive predictive value (PPV), recall or true positive rate (TPR) and F-measure of our proposed method. If we consider each pixel is independent object, precision is the probability of classified occluded pixel is true, while recall is the probability of detected occluded pixel is true. And F-measure is combined of PPV and TPR with the weight balance between them. Also, we measure the number of true positive (TP), true negative (TN), false positive (FP), false negative (FN) pixels to calculate these metrics above. The formula for calculating precision is:

$$PPV = \frac{TP}{TP + FP} \quad (10)$$

The recall (sensitivity) of each class is calculated by the following:

$$TPR = \frac{TP}{TP + FN} \quad (11)$$

And the F1-score is conducted from the following formula:

$$F_\beta = (1 + \beta^2) * \frac{PPV * TPR}{\beta^2 * PPV + TPR} \quad (12)$$

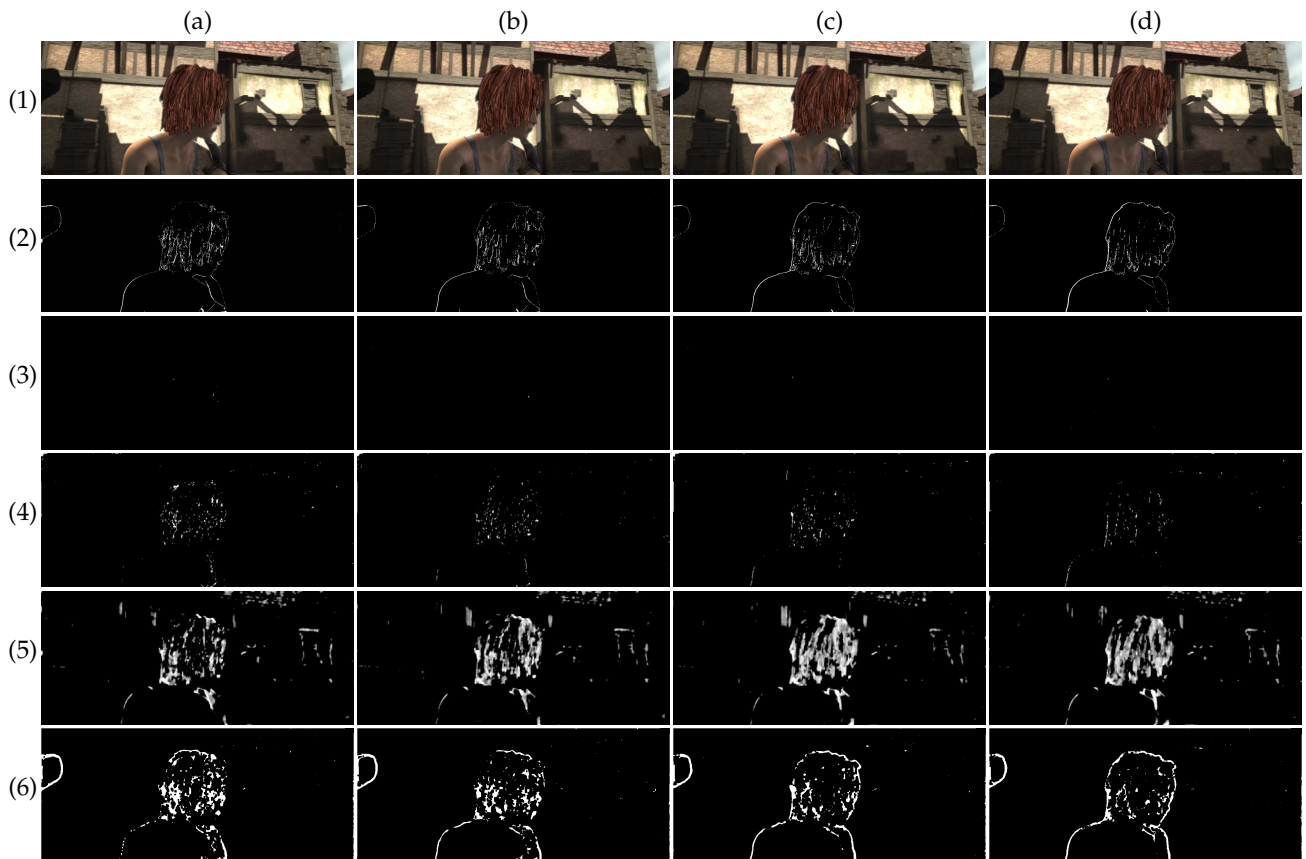


Figure 6. Visual Qualitative comparison on dataset *alley1*: (a) frame 33 (b) frame 34, (c) frame 35, (d) frame 36. From top to bottom present (1) input image frame, (2) ground truth, (3) Ayvaci *et al.* [13], (4) Alvarez *et al.* [8], (5) Estellers *et al.* [33], (6) Proposed method, respectively.

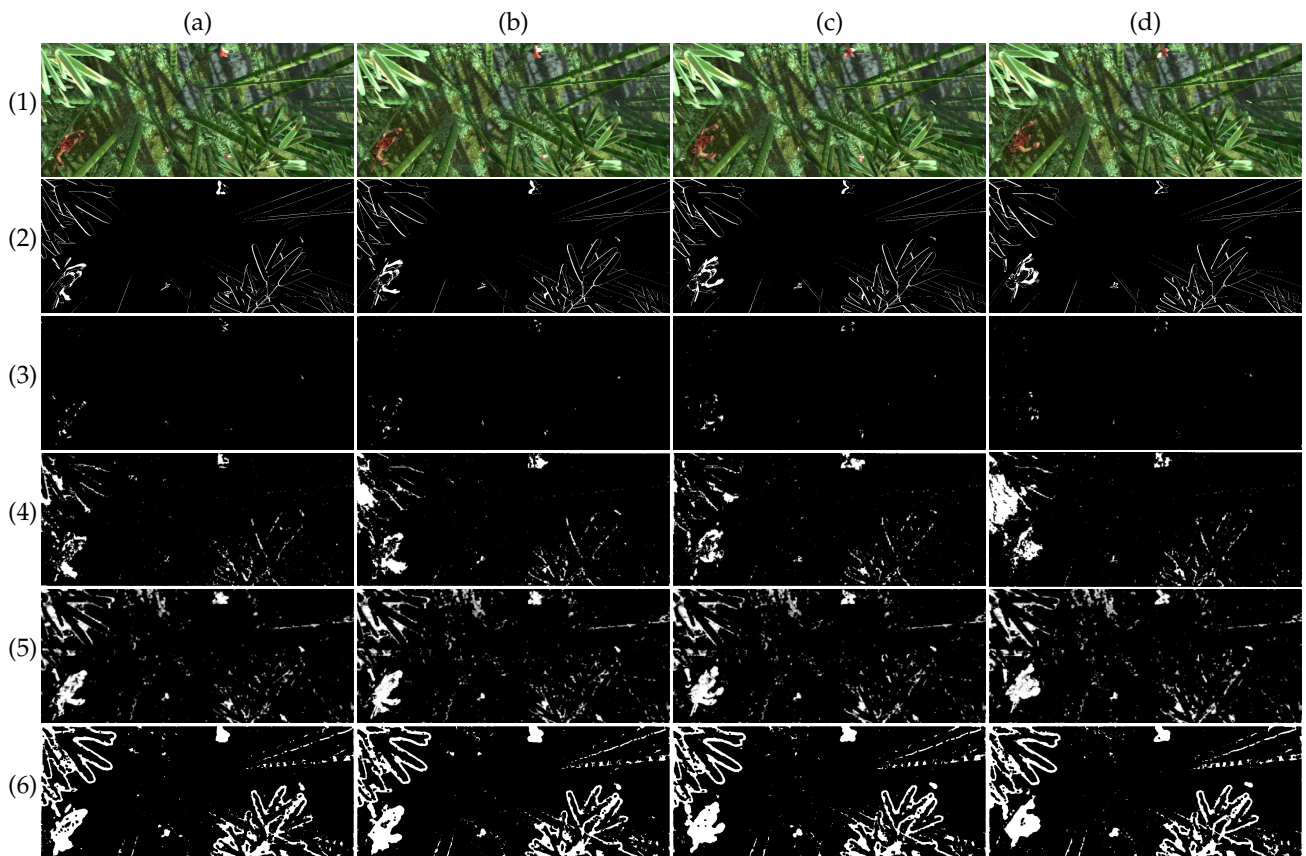


Figure 7. Visual Qualitative comparison on dataset *bamboo1*: (a) frame 01 (b) frame 02, (c) frame 03, (d) frame 04. From top to bottom present (1) input image frame, (2) ground truth, (3) Ayvaci *et al.* [13], (4) Alvarez *et al.* [8], (5) Estellers *et al.* [33], (6) Proposed method, respectively.

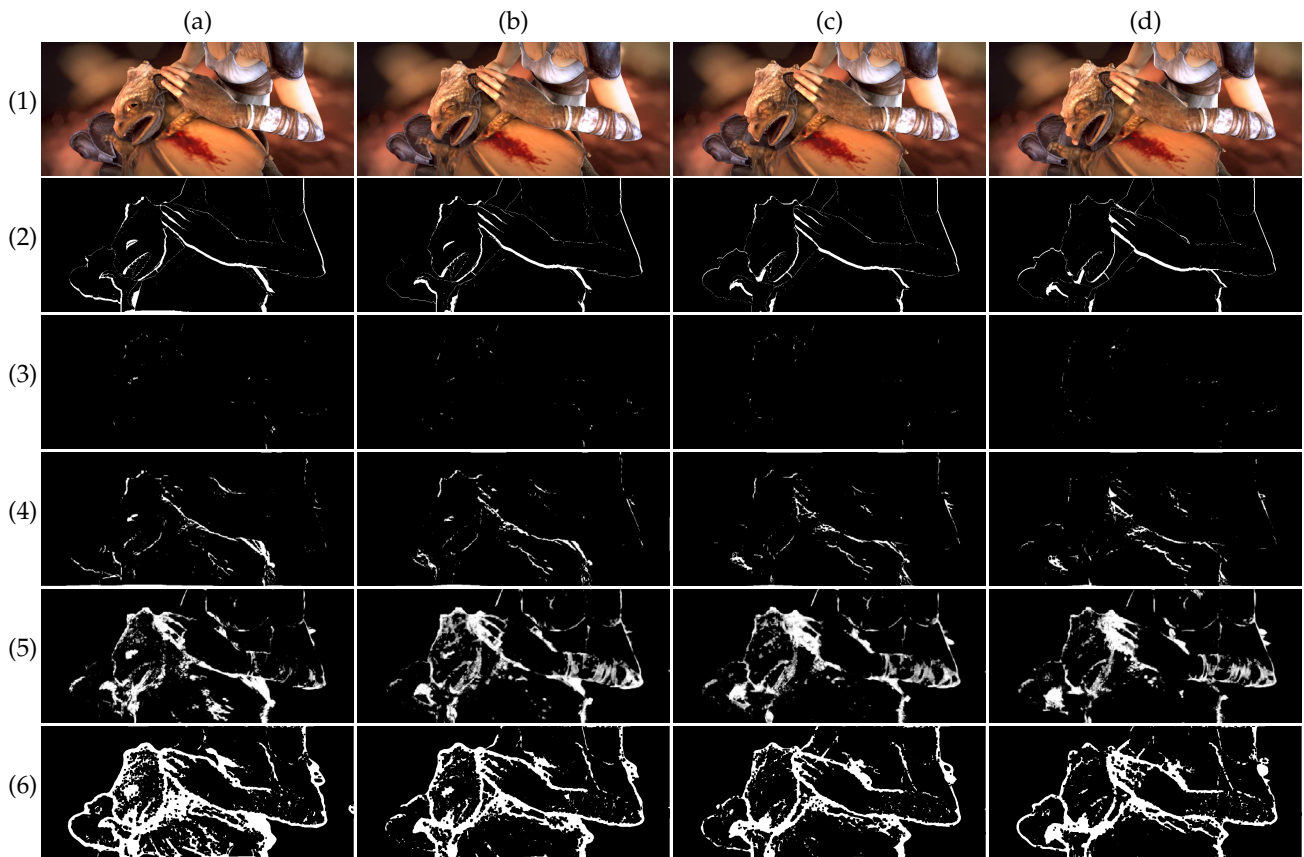


Figure 8. Visual Qualitative comparison on dataset *bandage1*: (a) frame 21 (b) frame 22, (c) frame 23, (d) frame 24. From top to bottom present (1) input image frame, (2) ground truth, (3) Ayvaci *et al.* [13], (4) Alvarez *et al.* [8], (5) Estellers *et al.* [33], (6) Proposed method, respectively.

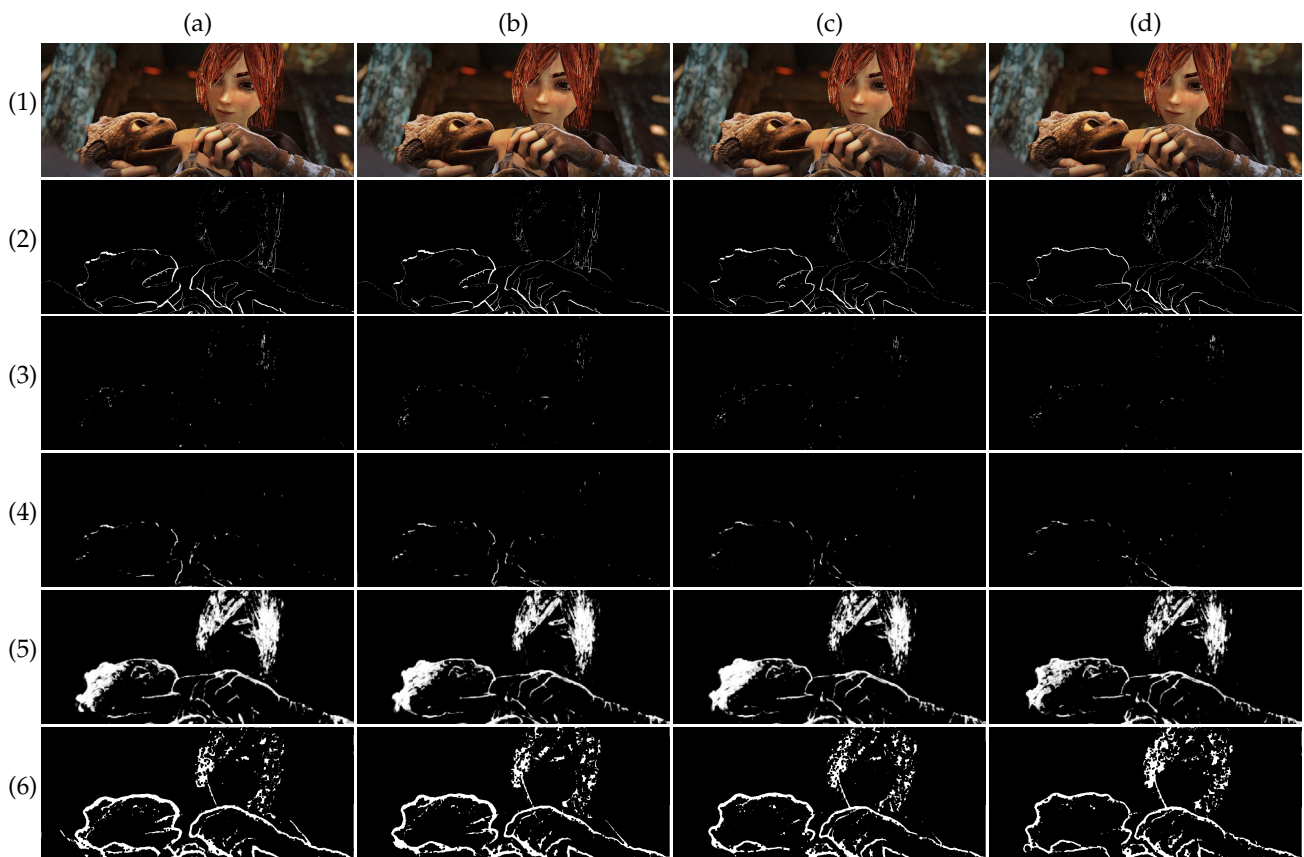


Figure 9. Visual Qualitative comparison on dataset *bandage2*: (a) frame 37 (b) frame 38, (c) frame 39, (d) frame 40. From top to bottom present (1) input image frame, (2) ground truth, (3) Ayvaci *et al.* [13], (4) Alvarez *et al.* [8], (5) Estellers *et al.* [33], (6) Proposed method, respectively.

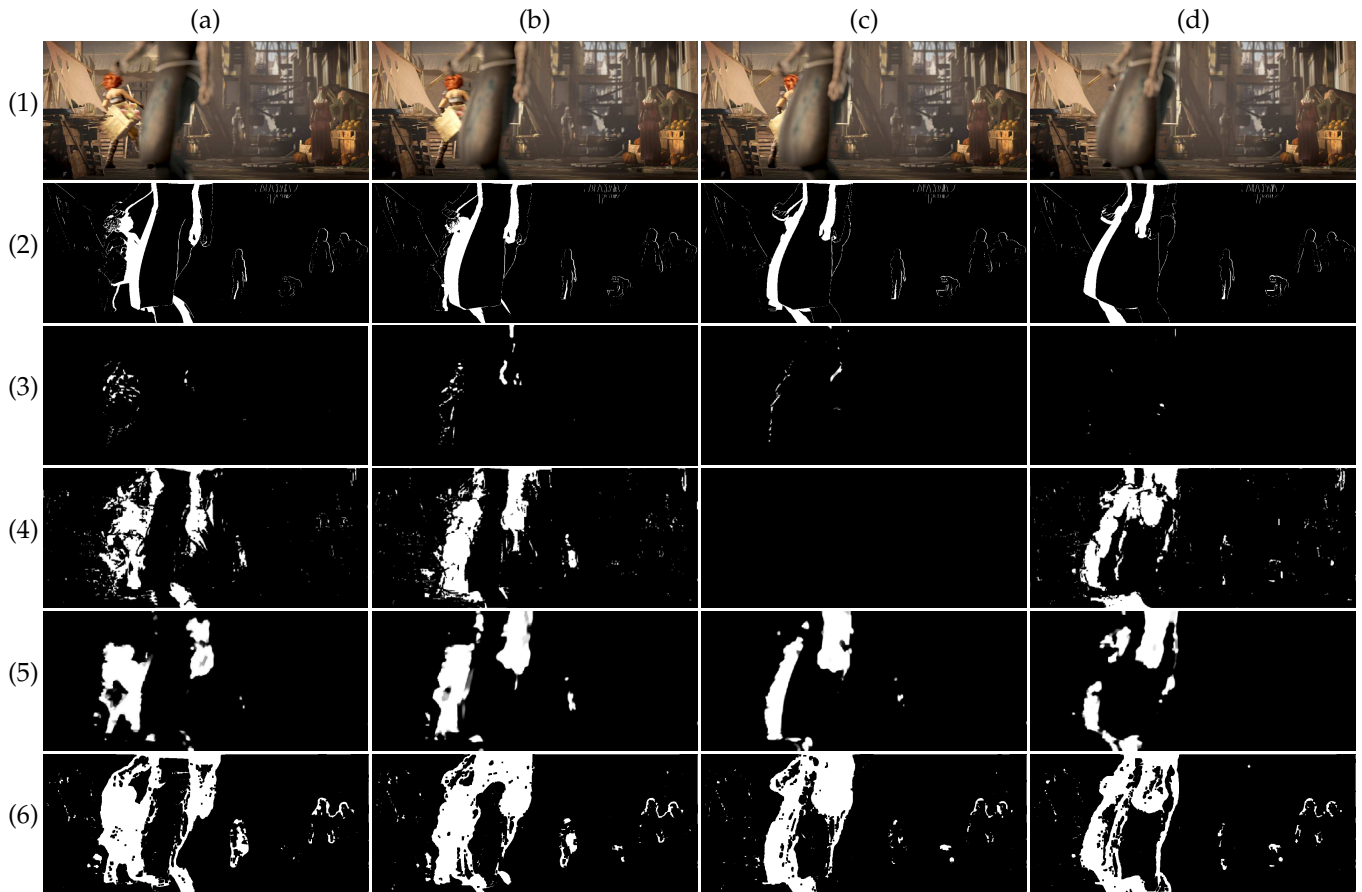


Figure 10. Visual Qualitative comparison on dataset *market2*: (a) frame 05 (b) frame 06, (c) frame 07, (d) frame 08. From top to bottom present (1) input image frame, (2) ground truth, (3) Ayvaci *et al.* [13], (4) Alvarez *et al.* [8], (5) Estellers *et al.* [33], (6) Proposed method, respectively.

Table I  
COMPARISON BETWEEN OUR METHOD AND EXISTING OCCLUSION DETECTION METHODS ALVAREZ *et al.*, AYVACI *et al.* AND ESTELLERS *et al.*  
OPTIMIZED FOR F1 SCORE

Sequence	Our method			Alvarez 2007			Ayvaci 2011			Estellers 2016		
	PPV	TPR	F1	PPV	TPR	F1	PPV	TPR	F1	PPV	TPR	F1
alley1	0.12	0.48	0.19	0.24	0.28	0.26	0.02	0.43	0.03	0.43	0.23	<b>0.29</b>
bamboo1	0.21	0.77	0.33	0.38	0.34	<b>0.36</b>	0.02	0.46	0.04	0.45	0.28	0.34
bandage1	0.19	0.88	0.31	0.17	0.33	0.22	0.01	0.39	0.01	0.45	0.28	<b>0.34</b>
bandage2	0.16	0.66	<b>0.26</b>	0.1	0.28	0.15	0.01	0.21	0.02	0.42	0.12	0.19
market2	0.35	0.88	<b>0.50</b>	0.46	0.34	0.38	0.02	0.47	0.04	0.32	0.37	0.32
Overall F1	0.32 <sup>(1)</sup>			0.27 <sup>(3)</sup>			0.03 <sup>(4)</sup>			0.30 <sup>(2)</sup>		

\* Bold denotes the best result for each sequence.

\*\* (1), (2), (3), (4) denotes the rank of each method in F1-score.

where,  $\beta$  is the parameter that controls the weighing balance between PPV and TPR. In this step, we use  $\beta = 1$  to make no extra weight to either PPV or TPR.

Table I compare our method to various techniques Alvarez *et al.* [8], Ayvaci *et al.* [13] and Estellers *et al.* [33]. Both Alvarez and Ayvaci compute the optical flow and occlusions between two consecutive frames, but Alvarez use the mismatch of forward and backward flows to compute the occlusion, while Ayvaci only uses the forward flow. Estellers proposes a hybrid method that integrates information from multiple frames to compute the occlusion detection such as

single-frame formulation from the previous frames, the spatial and temporal regularization eliminates isolated misdetections inconsistent in time or space.

### 4.3 Processing time

Finally, we tested the processing times of our method and other methods in Figure 11. Compared to the literature, our model is slightly slower than Ayvaci by 10%, but faster than the other two models Alvarez and Estellers by 200% and 18% respectively. Despite having a slower processing time, our results are more accurate



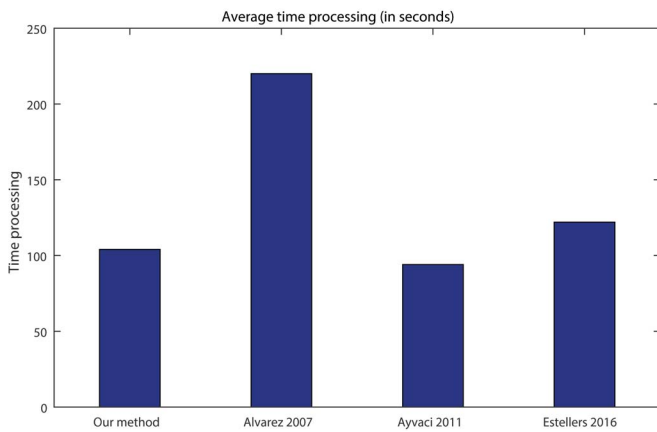


Figure 11. The average time processing of our method and other methods.

than Ayvaci. Therefore, these results demonstrate the efficiency of our model.

## 5 CONCLUSION

In this paper, we have presented the results of the work in process that can detect occlusion regions by using pixel-wise coherence, segment-wise confidence and interpolation technique. Our method gets better accuracy, timely processing and the coverage rate of almost occlusion regions than variation techniques Alvarez *et al.* [8], Ayvaci *et al.* [13] and Estellers *et al.* [33] in almost cases. It makes good shortcomings of these methods, such as objects move too fast, complex scene with many objects, background request be stationary, etc. Our method is a method which can get the same result as usual methods by solving only one PDE problem. Comparing with convex optimization methods, our algorithm gets a better result. Comparing with previous optimization methods, our algorithm is faster two times. Based on this results, future studies can extend to detect Occlusion regions better by using some visual features such as Flow & Reverse Flow Estimation, Gradient Magnitude, Texture Constancy, Temporal Gradient, Local & Global Flow Length Variance, etc.

## ACKNOWLEDGMENTS

This research is funded by Vietnam National University Ho Chi Minh City (VNU-HCM) under the grant number C2016-28-11/HD-KHCN

## REFERENCES

- [1] B. K. P. Horn, Ed., *Robot Vision*. Cambridge, MA, USA: MIT Press, 1986.
- [2] B. K. P. Horn and B. G. Schunck, "Determining optical flow," *Artificial Intelligence*, vol. 17, pp. 185–203, 1981.
- [3] T. Brox and J. Malik, "Large displacement optical flow: Descriptor matching in variational motion estimation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 3, pp. 500–513, Mar. 2011.
- [4] T. Pock, M. Unger, D. Cremers, and H. Bischof, "Fast and exact solution of total variation models on the GPU," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW'08)*. IEEE, Jun. 2008, pp. 1–8.
- [5] A. Wedel, T. Pock, C. Zach, H. Bischof, and D. Cremers, "Statistical and geometrical approaches to visual motion analysis." Berlin, Heidelberg: Springer-Verlag, 2009, ch. An Improved Algorithm for TV-L1 Optical Flow, pp. 23–45.
- [6] L. Xu, J. Jia, and Y. Matsushita, "Motion detail preserving optical flow estimation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 9, pp. 1744–1757, Sep. 2012.
- [7] P. Weinzaepfel, J. Revaud, Z. Harchaoui, and C. Schmid, "Deepflow: Large displacement optical flow with deep matching," in *Proceedings of the 2013 IEEE International Conference on Computer Vision*, ser. ICCV '13. Washington, DC, USA: IEEE Computer Society, 2013, pp. 1385–1392.
- [8] L. Alvarez, R. Deriche, T. Papadopoulou, and J. Sánchez, "Symmetrical dense optical flow estimation with occlusions detection," *International Journal of Computer Vision*, vol. 75, no. 3, pp. 371–385, Dec. 2007.
- [9] J. Xiao, H. Cheng, H. Sawhney, C. Rao, and M. Isnardi, "Bilateral filtering-based optical flow estimation with occlusion detection," in *Proceedings of the 9th European Conference on Computer Vision - Volume Part I*, ser. ECCV'06. Berlin, Heidelberg: Springer-Verlag, 2006, pp. 211–224.
- [10] S. Ince and J. Konrad, "Occlusion-aware optical flow estimation," *IEEE Transactions on Image Processing*, vol. 17, no. 8, pp. 1443–1451, Aug. 2008.
- [11] V. Kolmogorov and R. Zabih, "Computing visual correspondence with occlusions using graph cuts," in *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001*, vol. 2, 2001, pp. 508–515.
- [12] C. Strecha, R. Fransens, and L. Van Gool, *A Probabilistic Approach to Large Displacement Optical Flow and Occlusion Detection*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 71–82.
- [13] A. Ayvaci, M. Raptis, and S. Soatto, "Sparse occlusion detection with optical flow," *International Journal of Computer Vision*, vol. 97, no. 3, pp. 322–338, May 2012.
- [14] A. Humayun, O. M. Aodha, and G. J. Brostow, "Learning to find occlusion regions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, ser. CVPR '11. Washington, DC, USA: IEEE Computer Society, 2011, pp. 2161–2168.
- [15] M. Leordeanu, A. Zanfir, and C. Sminchisescu, "Locally affine sparse-to-dense matching for motion and occlusion estimation," in *Proceedings of the 2013 IEEE International Conference on Computer Vision*, ser. ICCV '13. Washington, DC, USA: IEEE Computer Society, 2013, pp. 1721–1728.
- [16] C. Ballester, L. Garrido, V. Lazcano, and V. Caselles, *A TV-L1 Optical Flow Method with Occlusion Detection*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 31–40.
- [17] M. J. Black and D. J. Fleet, "Probabilistic detection and tracking of motion boundaries," *International Journal of Computer Vision*, vol. 38, no. 3, pp. 231–245, Jul. 2000.
- [18] M. E. Sargin, L. Bertelli, B. S. Manjunath, and K. Rose, "Probabilistic occlusion boundary detection on spatio-temporal lattices," in *Proceedings of the IEEE 12th International Conference on Computer Vision*, Sept 2009, pp. 560–567.
- [19] A. N. Stein and M. Hebert, "Occlusion boundaries from motion: Low-level detection and mid-level reasoning," *International Journal of Computer Vision*, vol. 82, no. 3, pp. 325–357, May 2009.
- [20] A. Ayvaci and S. Soatto, "Motion segmentation with occlusions on the superpixel graph," in *Proceedings of the IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops*, Sept 2009, pp. 727–734.
- [21] P. Sundberg, T. Brox, M. Maire, P. Arbelaez, and J. Ma-

- lik, "Occlusion boundary detection and figure/ground assignment from optical flow," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, ser. CVPR '11. Washington, DC, USA: IEEE Computer Society, 2011, pp. 2233–2240.
- [22] X. He and A. Yuille, "Occlusion boundary detection using pseudo-depth," in *Proceedings of the 11th European Conference on Computer Vision: Part IV*, ser. ECCV'10. Berlin, Heidelberg: Springer-Verlag, 2010, pp. 539–552.
- [23] P. Smith, T. Drummond, and R. Cipolla, "Layered motion segmentation and depth ordering by tracking edges," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 4, pp. 479–494, Apr. 2004.
- [24] J. D. Jackson, A. J. Yezzi, and S. Soatto, "Dynamic shape and appearance modeling via moving and deforming layers," *International Journal of Computer Vision*, vol. 79, no. 1, pp. 71–84, Aug. 2008.
- [25] M. J. Black, "Layered segmentation and optical flow estimation over time," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, ser. CVPR '12. Washington, DC, USA: IEEE Computer Society, 2012, pp. 1768–1775.
- [26] J. Xiao and M. Shah, "Motion layer extraction in the presence of occlusion using graph cuts," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 10, pp. 1644–1659, Oct. 2005.
- [27] A. Ayvaci and S. Soatto, "Detachable object detection with efficient model selection," in *Proceedings of the 8th International Conference on Energy Minimization Methods in Computer Vision and Pattern Recognition*, ser. EMM-CVPR'11. Berlin, Heidelberg: Springer-Verlag, 2011, pp. 191–204.
- [28] C. Vogel, S. Roth, and K. Schindler, *An Evaluation of Data Costs for Optical Flow*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 343–353.
- [29] T. Brox, A. Bruhn, N. Papenberger, and J. Weickert, "High Accuracy Optical Flow Estimation Based on a Theory for Warping," in *Proceedings of the European Conference on Computer Vision*, 2004, vol. 4, no. May, pp. 25–36.
- [30] J. Shi and C. Tomasi, "Good features to track," in *Conference on Computer Vision and Pattern Recognition, CVPR 1994*, Seattle, WA, USA, Jun. 1994, pp. 593–600.
- [31] L. Xu, J. Chen, and J. Jia, "A segmentation based variational model for accurate optical flow estimation," in *Proceedings of the 10th European Conference on Computer Vision: Part I*, ser. ECCV '08. Berlin, Heidelberg: Springer-Verlag, 2008, pp. 671–684.
- [32] D. Han, "Comparison of Commonly Used Image Interpolation Methods," in *Proceedings of the 2nd International Conference on Computer Science and Electronics Engineering (ICCSEE 2013)*, no. Iccsee. Paris, France: Atlantis Press, 2013, pp. 1556–1559.
- [33] V. Estellers and S. Soatto, "Detecting occlusions as an inverse problem," *Journal of Mathematical Imaging and Vision*, vol. 54, no. 2, pp. 181–198, Feb. 2016.
- [34] D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black, "A naturalistic open source movie for optical flow evaluation," in *Proceedings of the 12th European Conference on Computer Vision - Volume Part VI*, ser. ECCV'12. Berlin, Heidelberg: Springer-Verlag, 2012, pp. 611–625.
- [35] M. Sokolova and G. Lapalme, "A systematic analysis of performance measures for classification tasks," *Information Processing and Management*, vol. 45, no. 4, pp. 427–437, 2009.
- [36] T. Fawcett, "An Introduction to ROC Analysis," *Pattern Recognition Letters*, vol. 27, no. 8, pp. 861–874, Jun. 2006.
- [37] D. M. W. Powers, "Evaluation: From precision, recall and f-measure to roc., informedness, markedness and correlation," *Journal of Machine Learning Technologies*, vol. 2, no. 1, pp. 37–63, 2013.
- [38] C. J. van Rijsbergen, *Information Retrieval*. Butterworth, 1979.



**Synh Viet-Uyen Ha** is lecturer at School of Computer Science and Engineering, International University - Vietnam National University HCMC, Vietnam. He received an engineer degree from Ho Chi Minh City University of Technology (HCMUT), Vietnam in 1996; and a M.Sc. degree from Ho Chi Minh University of Natural Science (HCMUNS), Vietnam in 1999, both in computer science and engineering. He completed his Ph.D. degree Electrical and Computer Engineering and postdoctoral research at School of Information and Communication Engineering, Sungkyunkwan University, Suwon, Korea in 2010. His research interests include computer machine, computer vision and image processing.



**Tuan-Anh Vu** is a master student and is completing a M.S. degree in Information Technology Management from International University, Vietnam National University, Ho Chi Minh City, Vietnam. He received an engineer degree from International University, Vietnam National University, Ho Chi Minh City, Vietnam in 2016. He has been doing research related to image processing and computer vision under the supervision of Dr. Synh Viet-Uyen Ha. His research interests include image processing, computer vision, artificial intelligence and robotics.



**Ha Manh Tran** is a lecturer of computer science at International University, Vietnam National University, Ho Chi Minh City, Vietnam. He received his master degree in 2004 from the University of Birmingham, United Kingdom and his doctoral degree in 2009 from Jacobs University Bremen, Germany. His research interests include network management, distributed computing, big data analytics.