A. Explanation of the Symbols

Table 4. Explanation of Symbols

Symbol	Definition
I	frames from the video
M	masks represented valid regions of each frame
$\hat{I}$	coarse aligned frames
$\hat{M}$	masks represented valid regions of each frame after coarse alignment
H	the height of each frame
W	the width of each frame
Op	the optical flow from coarse aligned frame to current frame
$\widetilde{Op}$	the reversed optical flow
$\widetilde{M}$	the reversed mask to denote the in boundary regions in coarse aligned frames
G	the edge extracted from current frames
$\hat{G}$	the edge extracted from coarse aligned frames
F	the features extracted by the encoder
D	the features generated by the decoder
κ	estimated affinity matrices
B	refined optical flow after propagation
$\hat{I}^e$	the extrapolated images
$\hat{M}^e$	the masks represented the valid regions after extrapolation
$\hat{G}^e$	the edges of extrapolated images

B. More Visual Comparison Results

In this section, we provide more visual comparison results by different methods with or without the use of our OVS method for video stabilization. In addition to the results of DUT [34], DIFRINT [5], Meshflow [21] and Yu *et al*. [38] provided in the main paper, we here provide the results by other two representative warping-based methods, *i.e*., PWStabNet [45] and StabNet [31], for a comprehensive comparison, as shown in Figure 7. It is clear that there is less content loss in the results of both PWStabNet and StabNet after incorporating OVS, demonstrating that OVS can effectively synthesize the out-of-boundary view to facilitate the warping processing in these methods. In addition, the distortion artifacts are also reduced, *e.g*., the building in the third column has less distortions with the help of OVS. Note that the results of PWStabNet and StabNet still have some content loss after stabilization since we only used the default 10 iterations in OVS to improve the cropping ratio. The content loss could be alleviated as the number of iterations increases.

We also provide a side-by-side comparison between DIFRINT [5] and our OVS based on DUT [34]. The results are shown in Figure 8. As can be seen, DIFRINT brings a zoom-in effect in the stabilized results, *e.g*., the tree in the first row and the tower in the third row are amplified while some content around the boundary are missed. On the contrary, our OVS can retain more content in the stabilized results, especially around the boundary. Besides,

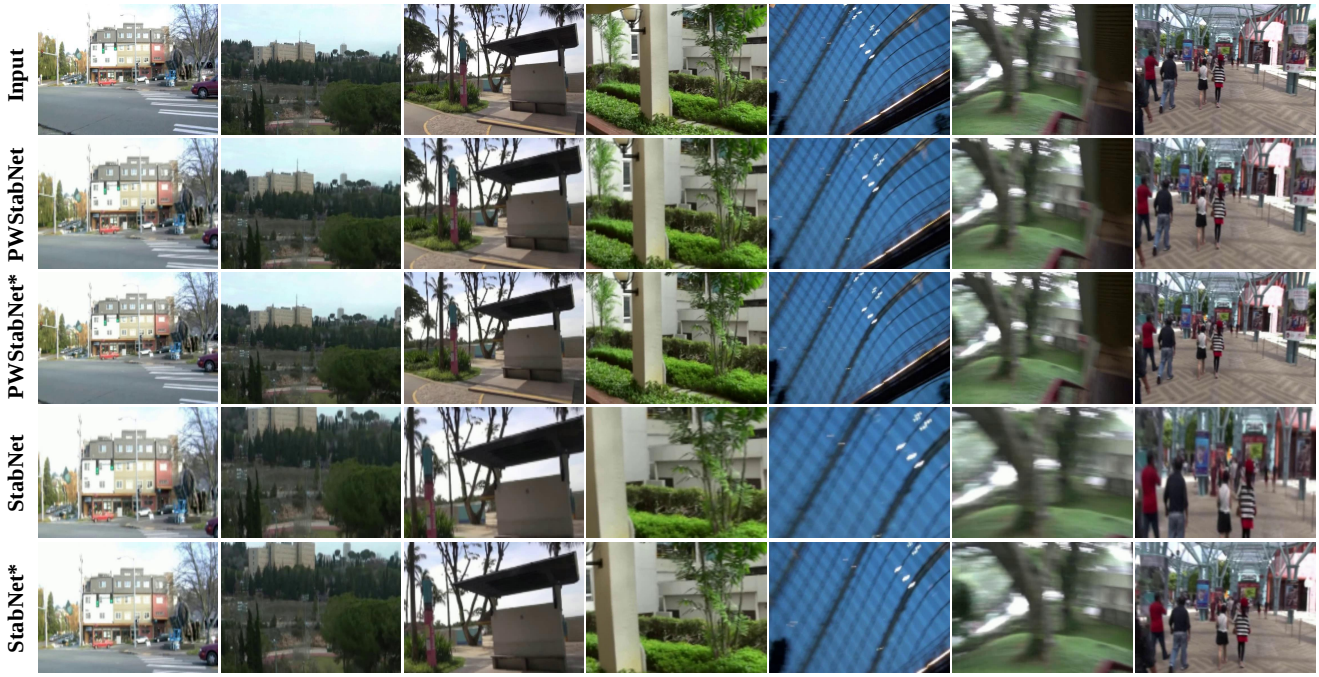

Figure 7. Visual comparison between PWStabNet [45] and StabNet [31]. * means stabilizers integrated with our OVS method.

	Regular			QuickRot			Parallax			Crowd			Running			Average		
	C	D	S	C	D	S	C	D	S	C	D	S	C	D	S	C	D	S
DIFRINT	0.939	0.934	0.838	0.953	0.633	0.826	0.946	0.922	0.817	0.939	0.910	0.809	0.938	0.871	0.825	0.943	0.854	0.823
PWStabNet	0.890	0.918	0.843	0.883	0.932	0.870	0.891	0.947	0.816	0.888	0.947	0.804	0.833	0.874	0.815	0.877	0.924	0.830
PWStabNet + OVS	0.968	0.965	0.846	0.952	0.934	0.869	0.958	0.961	0.819	0.959	0.961	0.806	0.957	0.961	0.818	0.959	0.957	0.831
Yu et.al	0.878	0.925	0.849	0.844	0.221	0.763	0.816	0.836	0.819	0.875	0.832	0.811	0.725	0.797	0.830	0.827	0.722	0.814
Yu et.al + OVS	0.968	0.939	0.852	0.861	0.362	0.845	0.929	0.876	0.826	0.930	0.899	0.811	0.920	0.845	0.835	0.922	0.784	0.834
StabNet	0.748	0.821	0.694	0.654	0.528	0.798	0.670	0.789	0.736	0.667	0.798	0.739	0.639	0.722	0.740	0.676	0.731	0.741
StabNet + OVS	0.818	0.875	0.697	0.738	0.699	0.815	0.755	0.872	0.736	0.733	0.829	0.742	0.772	0.868	0.752	0.763	0.829	0.749
Meshflow	0.751	0.891	0.851	0.819	0.302	0.783	0.773	0.681	0.799	0.751	0.776	0.784	0.757	0.715	0.848	0.770	0.673	0.813
MeshFlow + OVS	0.889	0.908	0.854	0.936	0.333	0.800	0.859	0.669	0.805	0.897	0.753	0.800	0.911	0.754	0.856	0.898	0.683	0.823
DUT	0.924	0.952	0.850	0.834	0.841	0.882	0.883	0.904	0.829	0.896	0.927	0.818	0.797	0.848	0.845	0.867	0.895	0.845
DUT + OVS	0.989	0.960	0.852	0.941	0.872	0.880	0.962	0.905	0.834	0.977	0.937	0.817	0.968	0.902	0.850	0.967	0.915	0.847
DUT + OVS*	0.999	0.981	0.853	0.998	0.929	0.881	0.998	0.943	0.837	0.999	0.953	0.818	0.999	0.918	0.854	0.999	0.944	0.849

Table 5. Category-wise objective metrics of different stabilizers. C, D, S are the abbreviations for Cropping ratio, Distortion, and Stability, respectively. "+OVS" means the stabilizer uses our OVS. * denotes the full-frame version of our OVS method.

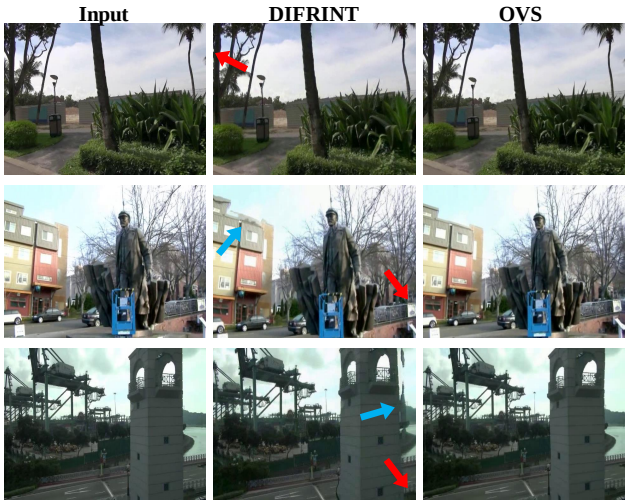

Figure 8. Side-by-side comparison between DIFRINT [5] and our OVS (based on DUT [34]). Red arrows indicate the content loss and blue arrows indicate the ghost artifacts.

DIFRINT produces obvious ghost artifacts in the results. We suspect that the interpolation process in DIFRINT uses inaccurate optical flow due to large jitter and object motion, resulting in ghost artifacts, especially around sharp edges and dynamic objects. In contrast, our OVS based DUT [34] have no such a drawback by exploiting the spatial coherence property in the video.

C. Category-wise Results of Objective Metrics

In Table 5, we provide the category-wise objective metrics of the warping-based stabilizers [21, 31, 45, 38, 34] and the interpolation-based one [5]. As can be seen, our OVS significantly improves the cropping ratio of the warping-based stabilizers on all categories. Besides, the stability and distortion metrics are also improved as a by-product of the increased cropping ratio.

D. Influence of Loss Weights

λ_1	λ_2	Cropping	Distortion	Stability
2	1.5	0.987	0.963	0.852
2	2.5	0.989	0.960	0.849
1.5	2	0.989	0.961	0.851
2.5	2	0.990	0.960	0.852
2	2	0.989	0.960	0.852

Table 6. Influence of loss weight.

We investigate the influence of the loss weights in the training objective. We rewrite the training objective function in the fine alignment stage as $L = L_I + \lambda_1 \times L_G + \lambda_2 \times L_M$, where λ_1 and λ_2 are set to 2 by default in our main experiments. In this experiment, we vary $\lambda_{1,2}$ from 1.5 to 2.5 and train the model for 200 epochs respectively, following the same training strategy and using the Adam optimizer. We report their results on the Regular category in the NUS dataset in Table 6. It can be seen that the performance of our model is not sensitive to the loss weights in terms of all the three metrics.

E. Analysis of the Cropping Ratio Issue in DIFRINT

DIFRINT is an interpolation-based full-frame video stabilizer that naturally avoids cropping after stabilization. Although there is no cropping, the result of DIFRINT still suffers from content loss, as shown in Figure 4 and Figure 8, suggesting that DIFRINT implicitly learns a zoom-in effect during stabilization. In other words, the cropping ratio of DIFRINT should be less than 1 even if it is a full-frame stabilizer by design. However, we notice that the cropping ratio of DIFRINT reported in some literature is 1. Here, we investigate the reasons behind this phenomenon. We use the official inference code, the pre-trained model, and the evaluation code provided by DIFRINT in the follow-

ing experiment. We find that there is a difference about the definition of the cropping ratio between DIFRINT [5] and Bundled [23]. In Bundled [23], which proposes the cropping ratio metric, the cropping ratio is defined as the reciprocal of the scale change between unstable and stable frames. However, DIFRINT calculates the reciprocal of the cropping ratio of all previous frames each time a new frame arrives in the evaluation code, as shown in Algorithm 1. It may cause some problems. For example, when a stabilized frame keeps only 50% of the original content after stabilization, the cropping ratio will be $1/0.5 = 2$. In this way, a frame with a large content loss will lead to a larger cropping ratio than a frame with a small content loss, which is unreasonable. Based on the above analysis, in all our experiments, we follow the exact definition in Bundled [23] as shown in Algorithm 2 for evaluation. The difference is highlighted by **red**.

Algorithm 1: Cropping Ratio Calculation in DIFRINT [5]

Input: Unstable frames: $\{f_i | i \in [1, E]\}$
Stabilized frames: $\{\widehat{f}_i | i \in [1, E]\}$;
Output: Cropping ratio: $\mathcal{C}$;
for $i = 1 : E$ **do**
  $H_i = Homography(f_i,\ \widehat{f}_i)$;
  $S_i = ScaleChange(H_i)$;
  $[\,\mathcal{C}\,] = Concat(1/[\,\mathcal{C}\,],\ S_i)$
end
$\mathcal{C} = min(mean([\,\mathcal{C}\,]),\ 1)$

Algorithm 2: Cropping Ratio Calculation in Bundled [23]

Input: Unstable frames: $\{f_i | i \in [1, E]\}$
Stabilized frames: $\{\widehat{f}_i | i \in [1, E]\}$;
Output: Cropping ratio: $\mathcal{C}$;
for $i = 1 : E$ **do**
  $H_i = Homography(f_i,\ \widehat{f}_i)$;
  $S_i = ScaleChange(H_i)$;
  $[\,\mathcal{C}\,] = Concat([\,\mathcal{C}\,],\ 1/S_i)$
end
$\mathcal{C} = mean([\,\mathcal{C}\,])$

F. User Study

To fully evaluate the performance of the proposed OVS method, we conduct a user study as a qualitative comparison. We select three videos from each category and compare the results of four stabilizers on these videos. Users are asked to rate them based on the overall visual experience. In addition, users are also required to rate them based on the visual experience in terms of content preserving. 28 male and female participants between the ages of 18 and 30 participated in this study. We average the scores of each stabilizer by category, and the results of the study are shown in Figure 9 and Figure 10. Adobe Premiere denotes the Warp Stabilizer with Synthesize Edge for stabilization in the commercial software. DIFRINT [5] is an interpolation-based stabilizer and DUT [34] is a SOTA warping-based stabilizer. The DUT stabilizer with the use of our OVS is denoted as OVS to avoid ambiguity. It is clear that DFIRINT outperforms the DUT and Adobe Premiere in terms of cropping ratio, but is slightly weaker in terms of overall visual experience because there are some ghost artifacts in its results. In terms of overall visual experience and crop ratio, our OVS method is the most preferred, demonstrating its superiority over existing methods.

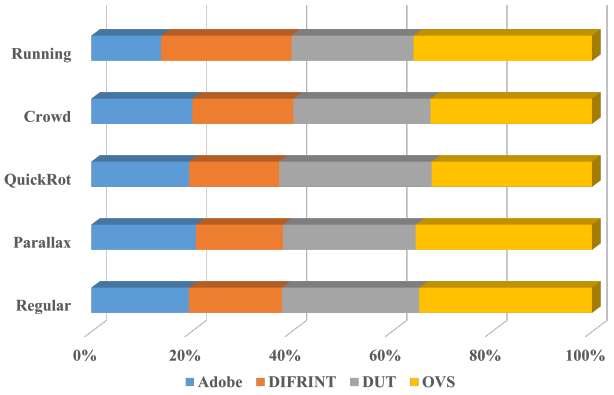

Figure 9. User study results of overall visual experience.

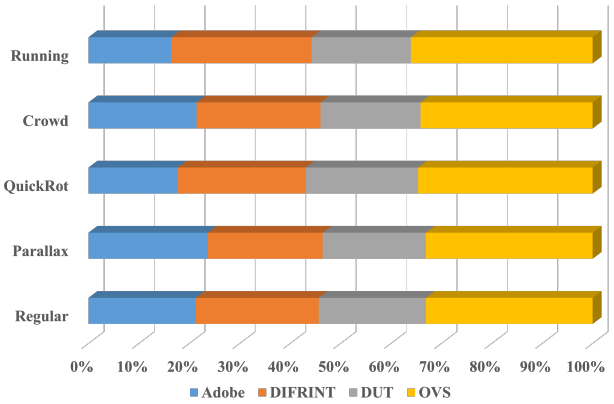

Figure 10. User study results of visual experience in terms of content preserving.