# Supplementary Material for "Enhancing Text-to-Video Editing with Motion Map Injection"

Anonymous ICCV submission

Paper ID 27

In this supplementary material, we describe related works, a method for calculating optical flow, an ablation study, a method for optical flow rotation, further explanation of the metric, and a github link to the code that could not be included in the extended abstract due to the lack of pages.

## 1. Related Work

### 1.1. Text-Guided Editing

The diffusion model [1, 2], which has recently been actively studied, generates data from noise through the process of adding or removing noise. Based on this diffusion model, text-guided image editing models such as DALL-E2 [3], Imagen [4], and stable diffusion [5] show the results of high-quality image editing. In particular, Prompt-to-Prompt [6] presents text-guided image editing that controls the relationship between the prompt text token and the corresponding image pixel with the attention maps, enabling unprecedented semantic editing. In addition, subsequent papers such as DreamBooth [7], EDICT [8], and Imagic [9] have been actively studied recently, showing impressive results for text-guided image editing.

Based on the significant progress of text-guided image editing, research has recently been expanded to text-guided video editing with the generative model. Dreamix [10] presents the first diffusion-based method of performing text-guided motion and application editing of videos through fine-tuning, but there are difficulties with localized editing by replacing a word. Video-P2P [11] divides their framework into two branches for unchanged parts and edited parts, and incorporates each attention map to enable detailed editing.

Concurrent to above works, vid2vid-zero [12] performs stable video reconstruction and editing by adding cross-frame attention to the U-Net structure of the existing diffusion model. In addition, FateZero [13], based on zero-shot, stores an attention map during the inversion process to maintain temporal consistency for structure and motion information. We attempt the first study to extract motion information directly from video and apply it to video editing.

### 1.2. Optical Flow Estimation

Optical flow estimation is a computer vision task that involves computing the motion of objects in a video sequence. Recently, this field is significantly advanced through the rise of deep neural networks. FlowNet [14] was the first fully convolutional neural network for estimating optical flow. Then, a series of works, represented by SpyNet [15], PWC-Net [16], LiteFlowNet [17], and RAFT [18] were proposed to reduce the computational costs through coarse-to-fine and iterative estimation methodology. Recently GMFlow [19] were proposed to achieve highly accurate results without relying on a large number of refinements by performing global matching with a Transformer.

Optical flow estimation is used in various video tasks. First, video action recognition [20, 21] aims to automatically recognize the behavior of objects in video sequences, where optical flow is used as a useful motion representation in video motion representation. Using spatio-temporal information from surrounding scenes to fill in new content for damaged areas, video inpainting [22, 23, 24] enables spatio-temporally stable synthesis between frames of video through optical flow. Video super resolution [25, 26, 27] is the field of generating high-resolution video frames from low-resolution video frames, and generally maintains temporal consistency between video frames by using optical flow as motion compensation. Video frame interpolation (VFI) [28, 29], a technology that generates an intermediate frame between two consecutive frames, also effectively extracts motion and shape information between frames by utilizing optical flow to estimate motion information between frames. Our work is the first attempt to apply optical flow to text-guided video editing where motion information is important based on the proven validity of the optical flow estimation in various video fields.

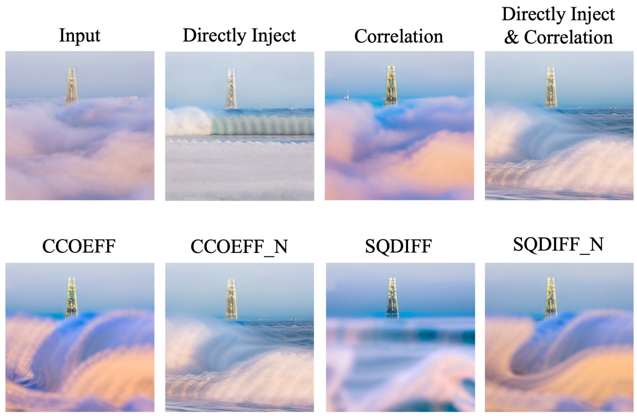


Figure 1. Outputs of various methods to measure the correlation between the motion map and the attention maps.

# 2. Ablation Study

Fig. 1 shows the results for different injection methods of motion map. "Directly Inject" injects the motion map directly into the attention map of the motion prompt. "Correlation" injects motion map to entire prompt's attention maps through the correlation between the each word attention maps and the motion map. "Directly Inject & Correlation" is combination of above described two methods. Specifically, "Correlation" is firstly applied, and then, "Directly Injection" is applied to the motion prompt. To inject the motion map into all words, the correlation between the attention maps of the input prompt and motion map is calculated. We applied various functions of template matching that represent a correlation between the two images. The "CCOEF" applied to the first image used correlation coefficient, and the second image is the result of "CCOEF_N" that normalizes it. The third image used "SQDIFF", a sum of squared differences, and the fourth image used "SQDIFF_N", which was normalized. As you can see in Fig. 1, the most suitable function to reinforce semantically editable is seen as "CCOEFF_N", which helps to edit semantically by varying weights depending on the degree of association of the prompter word. Therefore, we choose "CCOEFF_N" in method which is same as NCC.

# 3. Method for Optical Flow Rotation

Since $V_{flow}$ has information on the magnitude of pixel movement between frames, as well as the direction in which pixel moved between frames, the user can select and edit the motion value in the desired direction. Our model allows the user to edit contents in a specific direction by rotating the optical flow $V_{flow}$ according to the direction $D$ provided by the user before injecting it. We propose a method to edit using information on the direction in which pixels move in optical flow representing information on pixel motion between previous frame $F_{t-1}$ and current frame $F_t$. Our proposed model receives one of eight directions from the user, including Northeast (NE), Southeast (SE), Southwest (SW), and Northwest (NW), which are made from a combination of four directions in the 2D coordinate system.

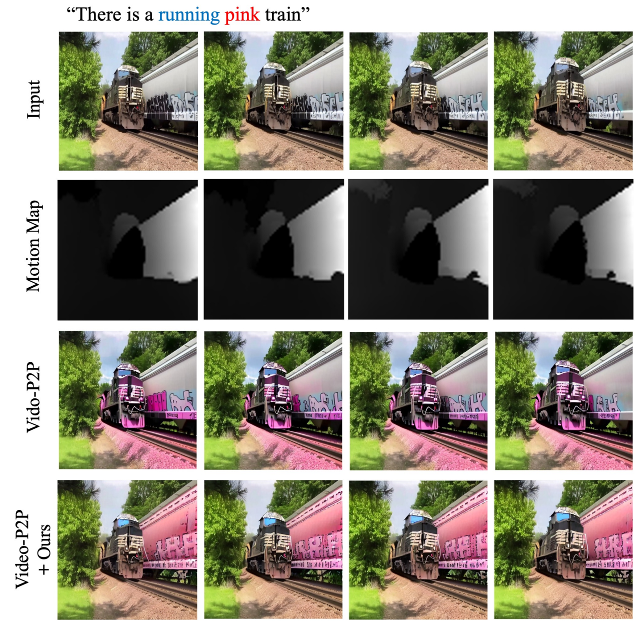


Figure 2. Editing method for objects moving in the direction specified by the user. Before editing, the user first selects one of the 8 directions.

$$\begin{bmatrix} X' \\ Y' \end{bmatrix} = \begin{bmatrix} X \cos\theta_D & -Y \sin\theta_D \\ X \sin\theta_D & Y \cos\theta_D \end{bmatrix}, \tag{1}$$

where $\theta_D$ denotes the angel between axis in frame and user provided direction $D$. $X, Y$ denotes each motion vector in axis $X$, and $Y$. $X', Y'$ denote the motion vectors rotated by $\theta$ in each axis.

After rotating the optical flow for the user-provided direction $D$, only the region of pixels with positive directional motion of the $x$ axis in the rotated coordinate system is specified. The value of the motion map is extracted from the specific region and video edit is performed on the corresponding area. The results can be seen in Fig. 2.

# 4. Detailed Description of Evaluation Metrics

**CLIP Score** [30] is calculated in the CLIP model [31], generating embedding vectors for input images and prompts. CLIP Score [30] is measured by computing the cosine similarity between image and caption embedding. We measured how close the target prompt and the edited video frames are semantically in the CLIP Score[30] . We measured the CLIP Score [30] between target prompt and each edited video frame, and quantitatively compared the performance of our model and other models by the average of the scores measured per each frame. The CLIP Score [30] is calculated with the following equation.

$$extCLIPScore(F_t, \mathcal{P}^*) = max(100*cos(E_{F_t}, E_{\mathcal{P}^*}), 0), \tag{2}$$

where $F_t$ denotes the $t$ th edited frame, and $\mathcal{P}^*$ denotes the target prompt. We use official ViT-Base-Patch16 CLIP model.

**Masked PSNR** To evaluate whether our proposed model performs undesired edit out of target region to be edited, we measured masked PSNR (M.PSNR) proposed by Video-P2P [11]. It indicates how much the external region of the target region has changed from the frame of the original video. In consideration of the averaged attention mask sequence $M$ of the changed object, we measure masked PSNR by computing the pixel distance in the out-of-target regions of the edited video $V^*$ and the input video $V$,

$$M.PSNR(V^*, V) = PSNR(B(V^*, M), B(V, M)), \tag{3}$$

according to Video-P2P [11], $B(V, M) = V_M$ is defined as a reversed mask binary function, so only regions not to be changed are involved in measuring masked PSNR.

## 5. Limitations

Accurate motion estimation of input video is essential for editing using optical flow. Therefore, even if optical flow is used, the bad results as shown in Fig. 3 may be obtained when it is difficult to estimate motion information from an image. The optical flow for the movement of fire could not be estimated, so there was no difference from the Video-P2P [11]. In addition, in the example of boat, motion was estimated only for ships that occupy a large area of the image, and small ships were not estimated. If the estimated motion of the optical flow for input video is not accurate, it is confirmed that our model, like existing Video-P2P [11], is difficult to perform accurate editing.

## 6. Code Descriptions

Our code is based on PyTorch version of Video-P2P [11]. We use Video-P2P [11] to edit videos. We set the parameters as follows: frame_size_h = 512, frame_size_w = 512, number of frames = 4,

Code is available at https://anonymous.4open.science/r/Motion_Map_Injection-4DF3/README.md

## 7. Additional Qualitative Results

Additional experimental results and code can be found in the supplementary archive zipped with the supplementary paper.

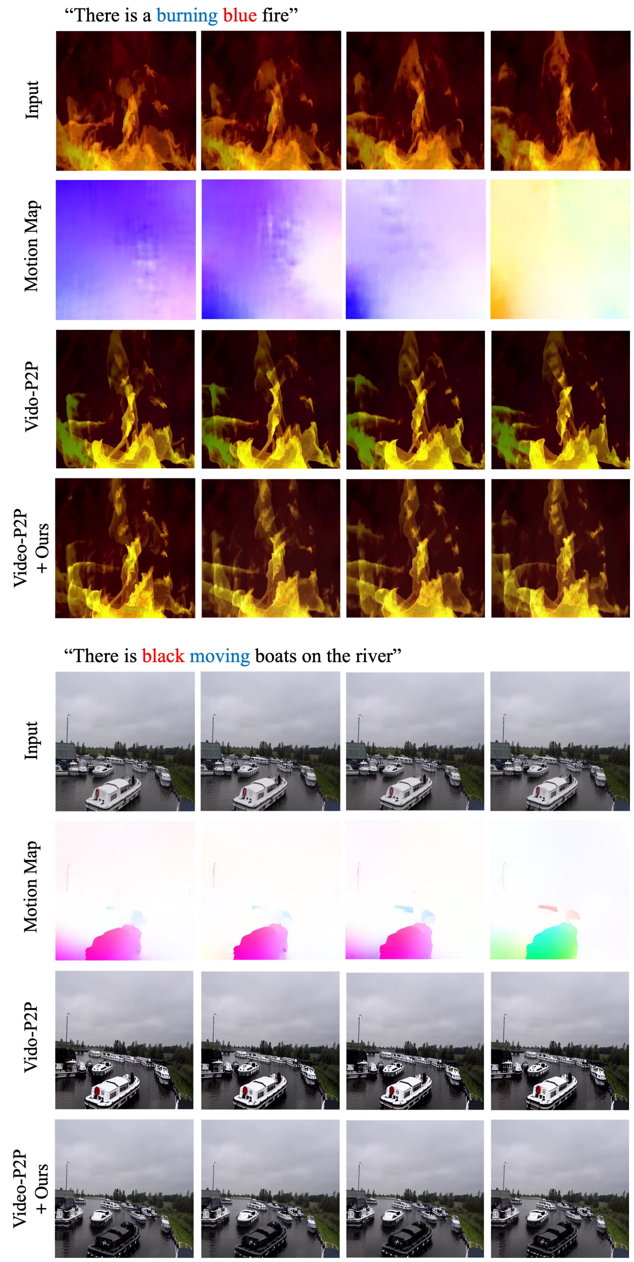


Figure 3. Results of Video-p2p [11] and our video editing model with inaccurately estimated optical flow

## References

[1] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020. 1

[2] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021. 1

[3] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image gen-

eration with clip latents. *arXiv preprint arXiv:2204.06125*, 2022. 1

[4] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems*, 35:36479–36494, 2022. 1

[5] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022. 1

[6] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-or. Prompt-to-prompt image editing with cross-attention control. In *The Eleventh International Conference on Learning Representations*, 2023. 1

[7] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22500–22510, 2023. 1

[8] Bram Wallace, Akash Gokul, and Nikhil Naik. Edict: Exact diffusion inversion via coupled transformations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22532–22541, 2023. 1

[9] Bahjat Kawar, Shiran Zada, Oran Lang, Omer Tov, Huiwen Chang, Tali Dekel, Inbar Mosseri, and Michal Irani. Imagic: Text-based real image editing with diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6007–6017, 2023. 1

[10] Eyal Molad, Eliahu Horwitz, Dani Valevski, Alex Rav Acha, Yossi Matias, Yael Pritch, Yaniv Leviathan, and Yedid Hoshen. Dreamix: Video diffusion models are general video editors. *arXiv preprint arXiv:2302.01329*, 2023. 1

[11] Shaoteng Liu, Yuechen Zhang, Wenbo Li, Zhe Lin, and Jiaya Jia. Video-p2p: Video editing with cross-attention control. *arXiv preprint arXiv:2303.04761*, 2023. 1, 3

[12] Wen Wang, Kangyang Xie, Zide Liu, Hao Chen, Yue Cao, Xinlong Wang, and Chunhua Shen. Zero-shot video editing using off-the-shelf image diffusion models. *arXiv preprint arXiv:2303.17599*, 2023. 1

[13] Chenyang Qi, Xiaodong Cun, Yong Zhang, Chenyang Lei, Xintao Wang, Ying Shan, and Qifeng Chen. Fatezero: Fusing attentions for zero-shot text-based video editing. *arXiv preprint arXiv:2303.09535*, 2023. 1

[14] Alexey Dosovitskiy, Philipp Fischer, Eddy Ilg, Philip Hausser, Caner Hazirbas, Vladimir Golkov, Patrick Van Der Smagt, Daniel Cremers, and Thomas Brox. Flownet: Learning optical flow with convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2758–2766, 2015. 1

[15] Anurag Ranjan and Michael J Black. Optical flow estimation using a spatial pyramid network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4161–4170, 2017. 1

[16] Deqing Sun, Xiaodong Yang, Ming-Yu Liu, and Jan Kautz. Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8934–8943, 2018. 1

[17] Tak-Wai Hui, Xiaoou Tang, and Chen Change Loy. Liteflownet: A lightweight convolutional neural network for optical flow estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8981–8989, 2018. 1

[18] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*, pages 402–419. Springer, 2020. 1

[19] Haofei Xu, Jing Zhang, Jianfei Cai, Hamid Rezatofighi, Fisher Yu, Dacheng Tao, and Andreas Geiger. Unifying flow, stereo and depth estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023. 1

[20] Shuyang Sun, Zhanghui Kuang, Lu Sheng, Wanli Ouyang, and Wei Zhang. Optical flow guided feature: A fast and robust motion representation for video action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1390–1399, 2018. 1

[21] Laura Sevilla-Lara, Yiyi Liao, Fatma Güney, Varun Jampani, Andreas Geiger, and Michael J Black. On the integration of optical flow and action recognition. In *Pattern Recognition: 40th German Conference, GCPR 2018, Stuttgart, Germany, October 9-12, 2018, Proceedings 40*, pages 281–297. Springer, 2019. 1

[22] Rui Xu, Xiaoxiao Li, Bolei Zhou, and Chen Change Loy. Deep flow-guided video inpainting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3723–3732, 2019. 1

[23] Dahun Kim, Sanghyun Woo, Joon-Young Lee, and In So Kweon. Deep video inpainting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5792–5801, 2019. 1

[24] Kaidong Zhang, Jingjing Fu, and Dong Liu. Flow-guided transformer for video inpainting. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XVIII*, pages 74–90. Springer, 2022. 1

[25] Zhigang Tu, Hongyan Li, Wei Xie, Yuanzhong Liu, Shifu Zhang, Baoxin Li, and Junsong Yuan. Optical flow for video super-resolution: a survey. *Artificial Intelligence Review*, 55(8):6505–6546, 2022. 1

[26] Mehdi SM Sajjadi, Raviteja Vemulapalli, and Matthew Brown. Frame-recurrent video super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6626–6634, 2018. 1

[27] Longguang Wang, Yulan Guo, Li Liu, Zaiping Lin, Xinpu Deng, and Wei An. Deep video super-resolution using hr optical flow estimation. *IEEE Transactions on Image Processing*, 29:4323–4336, 2020. 1

[28] Zhen Li, Zuo-Liang Zhu, Ling-Hao Han, Qibin Hou, Chun-Le Guo, and Ming-Ming Cheng. Amt: All-pairs multi-field transforms for efficient frame interpolation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9801–9810, 2023. 1

[29] Wenbo Bao, Wei-Sheng Lai, Chao Ma, Xiaoyun Zhang, Zhiyong Gao, and Ming-Hsuan Yang. Depth-aware video frame interpolation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3703–3712, 2019. 1

[30] Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. Clipscore: A reference-free evaluation metric for image captioning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7514–7528. Association for Computational Linguistics, nov 2021. 2

[31] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 2