# Supplementary Material: Learning-based Video Motion Magnification

Anonymous ECCV submission

Paper ID 12

## Summary of Contents

This is a part of the supplementary material. The contents of this supplementary material include the additional experiment results with descriptions and parameter setups, which have not been shown in the main paper due to the space limit.

The accompanied supplementary video contains comparisons with other methods [5,1,6,2] and baselines, self-evaluations and other applications.

## A.1 Parameters for Example Videos

In Table A.1, we specify parameters used in the experiments of the main paper such as magnification factors, and temporal filters. The parameters of examples in the supplementary video are self-contained if specified. For FIR filter, we chose the number of taps equal to the number of video frames in that sequence, and apply the filter in the frequency domain. For all other filters, we apply them in the time domain.

## A.2 Additional Experiments

This section describes the detail description of the dataset (in Sec. A.2.1), and presents additional results. We present an example for the characteristics of `Static` mode that magnifies broad frequency bands in Sec. A.2.2, qualitative comparison in Sec. A.2.3, quantitative analysis in Sec. A.2.4, visualization analysis examples in Sec. A.2.5, descriptions for other applications in Sec. A.2.6 and the supplementary video content in Sec. A.2.7.

### A.2.1 Detail dataset Information.

In the synthetic training data generation, we use two datasets: the MS COCO [26], and the PASCAL VOC segmentation dataset [25]. The license conditions are as follows: the annotations and images of the MS COCO are under Creative Commons Attribution 4.0 license and Flickr terms of use, respectively, and the PASCAL VOC is under Flickr terms of use and MSR Cambridge License (RTF).

All the real video examples used for comparisons come from either [5,6,1,9,2], otherwise our own captured data is used.

| Sequence Name | Magnification Factor | Temporal Band | Sampling Rate (fps) | Temporal Filter |
|---|---|---|---|---|
| crane | 75× | 0.2 – 0.25 Hz | 24 | FIR |
| balance | 10× | 1 – 8 Hz | 300 | $2^{nd}$ order Butterworth |
| throat | 100× | 90 – 110 Hz | 1900 | FIR |
| baby | 20× | See Temporal Filter Column | 30 | Difference of IIR, same as [6] |
| tree (high freq band) | 25× | 1.5 – 2 Hz | 60 | FIR |
| tree (low freq band) | 25× | 0.5 – 1 Hz | 60 | FIR |
| camera | 75× | 36 – 62 Hz | 300 | $2^{nd}$ order Butterworth |
| eye | 75× | 30 – 50 Hz | 500 | FIR |
| gun | 8× | N/A | N/A | `Dynamic` Mode |
| cat-toy | 7× | N/A | N/A | `Dynamic` Mode |
| drone | 10× | N/A | N/A | `Dynamic` Mode |
| hot-coffee | 3× | N/A | N/A | `Dynamic` Mode |
| drum | 10× | 74 – 78 Hz | 1900 | FIR |
| drum | 10× | N/A | N/A | `Static` Mode |
| guitar | 25× | 72 – 92 Hz | 600 | $2^{nd}$ order Butterworth |
| guitar | 10× | N/A | N/A | `Static` Mode |

**Table A.1.** Parameters of our results

### A.2.2 Comparison of temporal operations

Our network is trained on the two frames setting, but we also have shown that multi-frame linear temporal filtering is compatible to some extent. We discuss the difference of temporal operation characteristics for a reference purpose. Fig. A.1 shows the frequency response characteristics of different temporal operations.

Also, as shown in Fig. A.1, the area-under-covers of (a) are far broader than (b), *i.e.*, these modes magnify broader frequency ranges comparing to band-pass filtering in (b). This implies as follows. First, as the same amount of magnification factors are increased, overall energy increment of ours is even larger than the previous work; hence, the magnification factors are not directly comparable across modes, and this fact needs to be taken into account when the results are compared. Second, since noise (high) frequency band is also involved in, the training and working on the regime (a) would require more robust representation and synthesis power than the regime (b); thus, utilizing only two frames is much challenging regimes. For temporal filtering, magnifying a low-frequency band needs to take into account a long history of frames as shown in Fig. A.1-(c), while given an anchor frame, `Static` mode can magnify broad ranges of motion only with two frames, which is memory efficient.

We note that the subtraction operation at test phase is not claimed as the best, but they have trade-offs. The multi-frame based temporal filtering [5,1] has

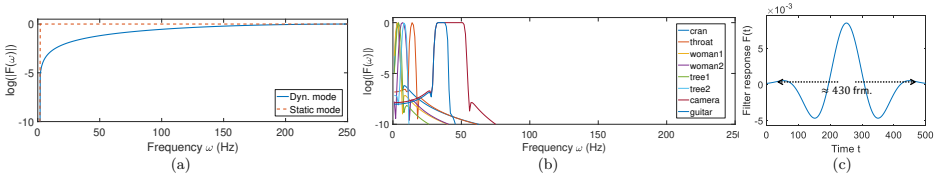



**Fig. A.1. Comparison on temporal operators.** The plots visualize log frequency responses of the subtraction operation for `Static` and `Dynamic` modes (a) and temporal filter examples (b) used in [5,1,6]. (c) temporal domain visualization of a low-freq. band-pass filter example (sampling rate: 60Hz, cut-off: [0.35, 0.71]Hz) used in [1], where the filter lies across near 450 frames. For visual comparison purpose, the frequency domain range is resampled to have the same range comparable between (a) and (b).

its own merits: selectivity of a frequency band of interest, and noise frequency band suppression as shown in Fig. A.1-(b). Since our network is trained only on (a), it is hard to expect any generalization for other temporal operations. Surprisingly, at test stage, we observed that our representation works favorably well with replacing the subtraction operation in the manipulator by the temporal filtering. Thus, we also conduct experiments with the subtraction operation to assess and compare the performance of our representation with the competing ones, *i.e.*, phase representations from complex steerable filters [1] and Riesz transformation [6] (see the supplementary video). Also, we compare our method with the competing methods equipped with temporal filtering.

Fig. A.2 shows its example that `Static` mode with our method magnifies the motion of all three strings whose frequencies are different. This shows that our method magnifies true motions, but not hallucinating.

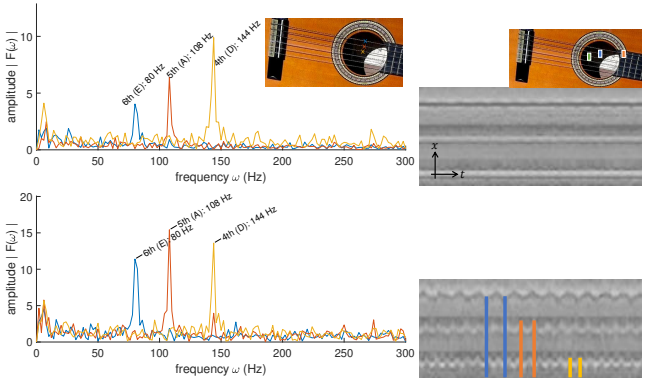



**Fig. A.2. Broad frequency magnification (Guitar sequence).** The `Static` mode with our method does not alter input frequency, as well as magnifies overall the frequency. The estimated frequencies of $4, 5, 6$-th strings are $\{137.14, 109.10, 80\}$-Hz, which is very close values to ideal frequencies, $\{144, 108, 80\}$-Hz. Color lines on the slice view are the samples of the period measure to estimate frequency.

### A.2.3 Additional qualitative Comparisons.

Various qualitative results can be found in the supplementary video. *Due to the characteristic of the problem, we strongly recommend to refer to the supplementary video for complete comparison and evaluation.*

We also present another comparison with other methods in Fig. A.3, where we replace the all the temporal operation to be `Dynamic` mode, so that we can compare the capability of the representation and synthesis of each method under the same setting.

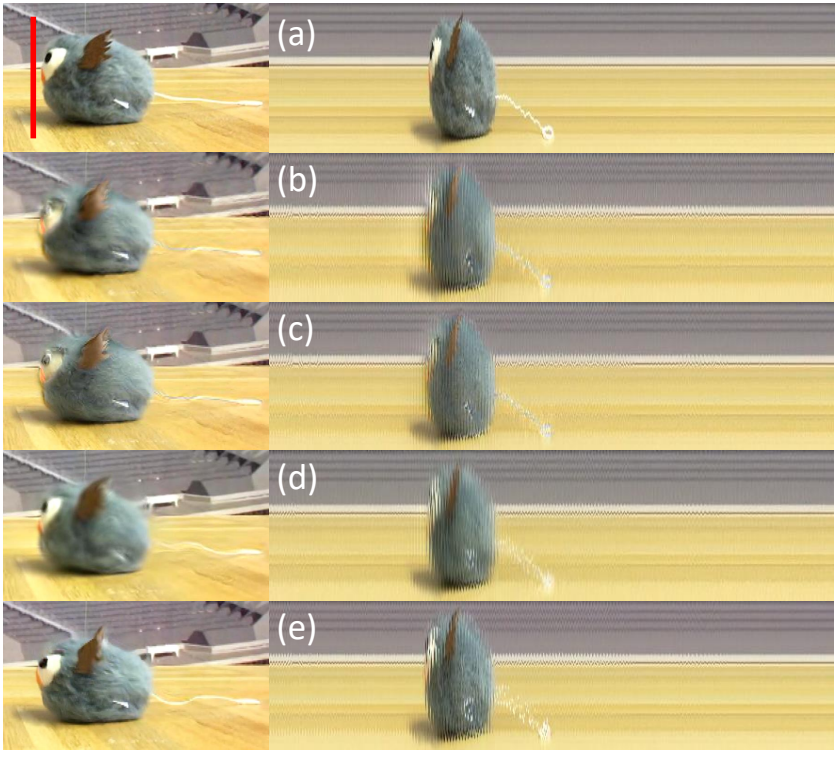


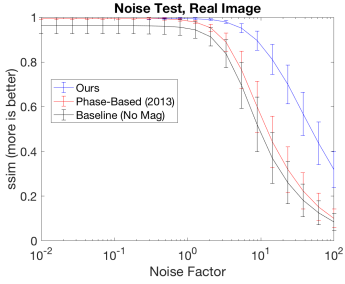

**Fig. A.3. Slice view comparison with other methods, `Dynamic` mode.** We compare original inputs (a) for reference, Wadhwa *et al.* [1] (b), Wadhwa *et al.* [6] (c), Zhang *et al.* [2] (d) and the proposed method (e). Even though there are large translational motions, only our method shows clear boundaries with proper magnified motion in slice view, while the other methods have blurry textures due to artifacts.

### A.2.4 Additional Quantitative Comparisons.

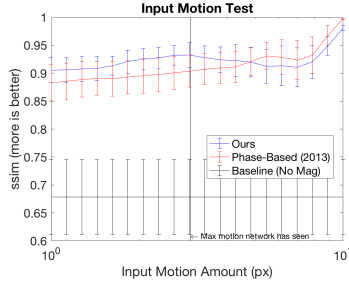
We summarize the test parameters used in all quantitative results in Table A.2. In Fig. A.4, we show additional quantitative evaluations for completeness: noise

**Table A.2. Summary of parameter for each quantitative test.**

| Param. \ Test | Magnification Factor | Motion (px.) | Magnified Motion (px.) | Noise | Background Motion |
|---|---|---|---|---|---|
| Sub-pixel | Capped at 100 | 0.01 - 1 | Capped at 10 | No | Yes |
| Noise | 5.0 | 2.0 | 10.0 | 0.01 - 100 | No |
| Noise on Real Image | 2.0 (estimate) | - | - | 0.01 - 100 | No |
| Magnification Factor | 1 - 1,000 | 0.01 - 2.0 | Capped at 10 | No | Yes |
| Input Motion | Capped at 5.0 | 1 - 10 | Capped at 10 | No | Yes |
| Magnified Motion | Capped at 5.0 | 1 - 10 | 1 - 30 | No | Yes |



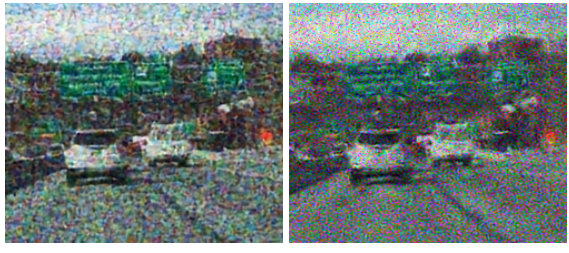

(a) Noise performance in real images




(b) Input motion test. Because magnified motion is capped at 10px, magnification factor approaches 1 towards the end, which explain the performance.

**Fig. A.4. Additional Quantitative Analysis**.

performance on real data and input motion range test. To test the noise performance on real examples in (a), we grab three consecutive frames from 17 high speed videos, and approximate their motions as linear. That is, the third frame will have twice the amount of motion as the second frame. We synthetically add noise as described before.



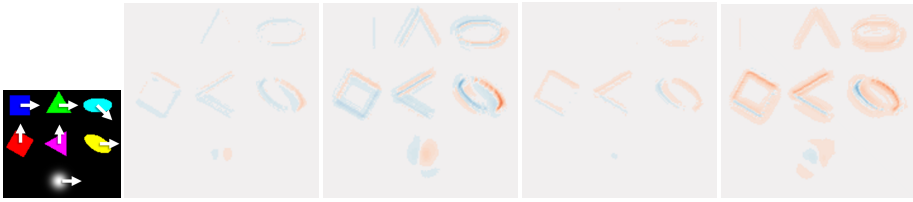
(a) (b)

**Fig. A.5. Example output from extreme noise test on real images (Noise factor = 9)**. (a) Our method preserves the detail better and results in less noise than (b) the phase-based method [1].

To see the extreme behavior against noise, we qualitatively compare the cases in Fig. A.5. Our method tends to preserve detail better (notice the car) rather than blurring, and have a better structure preserving quality in the output than

the phase-based one. Since our filters are directly learned in a data-driven way, we can easily augment the data to induce desired properties.

### A.2.5 Additional Network Visualization for Analysis



(a) Unit A Unit B

(b) $\alpha$=2 (c) $\alpha$=5 (d) $\alpha$=2 (e) $\alpha$=5

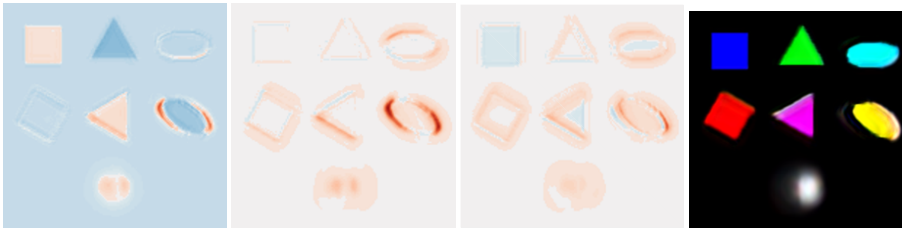
**Fig. A.6. Visualization of activations at Probe pt. B in Fig. 2, *i.e.*, $h(\alpha(g(\mathbf{A}_2 - \mathbf{A}_1))$.** (a) Example input, where the arrows indicate the motion directions of each shape objects. Comparing (b,d) to (c,e), as the magnification factors are increased, the manipulator layer not only increases the scale of activation, but also propagates the activation around. In this regard, our network seems to learn how to compensate object movements according to resulting motion.

In Fig. A.6-(b–e), we compare the activations of the compensation part, *i.e.*, $h(\alpha(g(\mathbf{A}_2 - \mathbf{A}_1))$, with two different magnification factors $\alpha$ for the same input motion. It shows that, according to $\alpha$, the compensation not only scales activations but also spatially propagates around. This suggests that our representation acts as a proxy of motion information that is able to be interpreted and manipulated. Lastly, observing activations of the compensation part in Fig. A.6 and the fusion layer in Fig. A.7, the network seems to learn how to compensate discrepancy introduced by motion manipulation, rather than explicitly moving pixels. These suggest that our method learns a representation closely related to Eulerian motion representation rather than Lagrangian one.

### A.2.6 Applications

To show the versatility of our learned motion representation, we additionally present potential applications, view/frame interpolation. Note that we directly use the network trained on our synthetic data for motion magnification, and we have never re-trained or fine-tuned the network for the specific applications. The results can be found in the supplementary video.

For the view/frame interpolation applications, we use our network to generate intermediate frames between two input frames by changing magnification factor in a range of [0,1]. For the view interpolation example, we generate and insert 25 intermediate frames given two input images. For the frame interpolation example, we generate five frames between every two consecutive frames, *i.e.*, 5× temporal interpolation. We use the light field camera dataset [28] in these applications.



(a) Unit02 (b) Unit13 (c) Unit19 (d) Decoded

**Fig. A.7. Visualization of activations at Fusion layer in Fig. 2.** We show randomly sampled activations fused from visual and motion compensated representations. The representations and decoded frame show that our network compensates discrepancy induced by motion magnification visually, rather than explicit pixel movement. It shows a color synthesis behavior of our network in (d). Around the yellow ellipse in (d), the color values seem to be synthesized to compensate movements in a similar way to inpainting.

### A.2.7 Supplementary Video Content

The content summary of the supplementary video is as follows:

- Qualitative comparison with temporal filter
- Qualitative comparison in 2 frame input setting (static mode, dynamic mode, frequency characteristics comparison)
- Additional analysis
- Applying different magnification factors without re-training (including motion attenuation examples)
- Applications: View/Frame interpolation.

# References


1. Wadhwa, N., Rubinstein, M., Durand, F., Freeman, W.T.: Phase-based video motion processing. ACM Trans. Graph. **32**(4) (2013) 80
2. Zhang, Y., Pintea, S.L., van Gemert, J.C.: Video Acceleration Magnification. In: IEEE Conf. on Comput. Vis. and Pattern Recognit. (2017)
3. Cha, Y.J., Chen, J., Büyüköztürk, O.: Output-only computer vision based damage detection using phase-based optical flow and unscented kalman filters. Engineering Structures **132** (2017) 300–313
4. Balakrishnan, G., Durand, F., Guttag, J.: Detecting pulse from head motions in video. In: IEEE Conf. on Comput. Vis. and Pattern Recognit. (2013) 3430–3437
5. Wu, H.Y., Rubinstein, M., Shih, E., Guttag, J., Durand, F., Freeman, W.: Eulerian video magnification for revealing subtle changes in the world. ACM SIGGRAPH **31**(4) (2012) 65–8
6. Wadhwa, N., Rubinstein, M., Durand, F., Freeman, W.T.: Riesz pyramids for fast phase-based video magnification. In: IEEE Int. Conf. on Comput. Photogr. (2014)
7. Freeman, W.T., Adelson, E.H.: The design and use of steerable filters. IEEE Trans. Pattern Anal. Mach. Intell. **13**(9) (1991) 891–906
8. Liu, C., Torralba, A., Freeman, W.T., Durand, F., Adelson, E.H.: Motion magnification. ACM Trans. Graph. **24**(3) (2005) 519–526
9. Elgharib, M.A., Hefeeda, M., Durand, F., Freeman, W.T.: Video magnification in presence of large motions. In: IEEE Conf. on Comput. Vis. and Pattern Recognit. (2015)
10. Kalantari, N.K., Wang, T.C., Ramamoorthi, R.: Learning-based view synthesis for light field cameras. ACM SIGGRAPH Asia **35**(6) (2016) 193–10
11. Wang, T., Zhu, J., Kalantari, N.K., Efros, A.A., Ramamoorthi, R.: Light field video capture using a learning-based hybrid imaging system. ACM Trans. Graph. **36**(4) (2017) 133:1–133:13
12. Liu, Z., Yeh, R.A., Tang, X., Liu, Y., Agarwala, A.: Video Frame Synthesis using Deep Voxel Flow. In: IEEE Int. Conf. on Comput. Vis. (2017)
13. Niklaus, S., Mai, L., Liu, F.: Video Frame Interpolation via Adaptive Convolution. IEEE Conf. on Comput. Vis. and Pattern Recognit. (2017)
14. Niklaus, S., Mai, L., Liu, F.: Video Frame Interpolation via Adaptive Separable Convolution. In: IEEE Int. Conf. on Comput. Vis. (2017)
15. Mathieu, M., Couprie, C., LeCun, Y.: Deep multi-scale video prediction beyond mean square error. Int. Conf. on Learn. Representations (2016)
16. Srivastava, N., Mansimov, E., Salakhudinov, R.: Unsupervised learning of video representations using lstms. In: Int. Conf. on Mach. Learn. (2015)
17. Villegas, R., Yang, J., Hong, S., Lin, X., Lee, H.: Decomposing motion and content for natural video sequence prediction. In: Int. Conf. on Learn. Representations. (2017)
18. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: IEEE Conf. on Comput. Vis. and Pattern Recognit. (2015)
19. Sajjadi, M.S., Schölkopf, B., Hirsch, M.: Enhancenet: Single image super-resolution through automated texture synthesis. IEEE Int. Conf. on Comput. Vis. (2017)
20. Radford, A., Metz, L., Chintala, S.: Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv preprint arXiv:1511.06434 (2015)
21. Odena, A., Dumoulin, V., Olah, C.: Deconvolution and checkerboard artifacts. Distill **1**(10) (2016) e3

22. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution. In: Eur. Conf. on Comput. Vis., Springer (2016) 694–711
23. Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. IEEE Conf. on Comput. Vis. and Pattern Recognit. (2017)
24. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. CoRR **abs/1412.6980** (2014)
25. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The pascal visual object classes (voc) challenge. Int. J. of Comput. Vis. **88**(2) (June 2010) 303–338
26. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: Eur. Conf. on Comput. Vis., Springer (2014) 740–755
27. Bau, D., Zhou, B., Khosla, A., Oliva, A., Torralba, A.: Network dissection: Quantifying interpretability of deep visual representations. In: IEEE Conf. on Comput. Vis. and Pattern Recognit. (2017)
28. Wilburn, B., Joshi, N., Vaish, V., Talvala, E.V., Antunez, E., Barth, A., Adams, A., Horowitz, M., Levoy, M.: High performance imaging using large camera arrays. In: ACM Transactions on Graphics (TOG). Volume 24., ACM (2005) 765–776