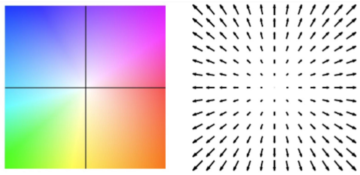
Color-coding for visualizing motion vector



Supplementary Materials - Modeling Human Visual Motion Processing with Trainable Motion Energy Sensing and a Self-attention Network

Anonymous Author(s)
Affiliation
Address
`email`

We have included some of the implementation information in the supplementary material. Figures referenced in the supplementary material correspond to those presented in the main text. Furthermore, to enhance the visual representation of our work, we have created videos showcasing select stimuli. For a concise overview, we recommend accessing the quick links embedded in the PDF. In case the quick links are unavailable, one can directly refer to the attached videos.

1 Training Details

1.1 Dataset

Current methods for estimating optical flow using DNNs can be categorized as unsupervised/self-supervised and supervised learning approaches. While unsupervised learning methods are intuitively similar to creatures' interaction with the world, most current methods based on differentiable image warping[1–3] still attempt to approximate physical motion GT. Therefore, we choose to use the supervised learning approach, which is more straightforward as recent research suggests that human perception of motion is highly similar to physical GT[4].

To train and evaluate the model, we construct a dataset containing various natural and artificial motion scenes. Specifically, we incorporate the Sintel benchmark[5], the DAVIS[6] dataset with pseudo-labels generated by FlowFormer[7], as well as self-created multi-frame datasets with non-texture motions and drifting grating motion. [1]

Simple non-texture motion: As the name suggests, non-texture motion refers to scenarios where both the moving object and the background exhibit low texture density. We incorporate non-texture motion as part of our training data due to the incompleteness of existing optical flow datasets. Our observations show that current DNN models exhibit suboptimal generalization performance on simple stimuli commonly employed in vision research, such as a drifting grating and a simple moving box. (Check here and here for video demos of unstable results of official RAFT[8]).

There are various explanations for this phenomenon. First, non-texture stimuli, such as drifting gratings, present multiple ambiguous motion solutions, which may confuse the model. Additionally, the model may not have been trained on non-texture datasets, leading to difficulties in generalizing to such stimuli. Consequently, the model's instability causes significant deviations from human interpretation in such pure experimental scenarios, hindering the study of the model using stimuli commonly employed in vision science.

To alleviate the problem, we used a concise method to construct a non-texture dataset. We first employed a superpixel algorithm [9] to extract the self-regions of natural images, followed by a low-pass filter to smooth out the texture. We practiced affine transformations to simulate two-dimensional translations, rotations, deformations, etc., of the object and inverse affine transformations to track the optical flow. The motion of each step is sampled from a random Gaussian distri-

---

[1] The proposed dataset and the project code will be publicly available online once the work get published.

Submitted to 37th Conference on Neural Information Processing Systems (NeurIPS 2023). Do not distribute.

bution, and we use a Markovian stochastic process to simulate inter-frame smoothed motion over multiple frames, where the current moment's motion is correlated with the previous motion state. Specifically, we assume that random object movement $\mathbf{S}(t)$ can be decomposed into two orthogonal vectors $\mathbf{S}(t) = [U(t), V(t)]$. For any time step t, we introduce Markov properties by assuming that the motion state depends only on the motion in the previous time step $t-1$:

$$\Pr\left[\mathbf{S}(t)\right] = \Pr[\mathbf{S}(t) = s_t \mid \mathbf{S}(t-1) = s_{t-1}] \tag{1}$$

The motion states $[U, V]$ at time t are sampled from the 2D Gaussian distributions with condition of previous motion state. Check here for a demo of the non-texture training sample.

Drifting Grating: Drifting gratings are commonly used stimuli in visual studies, which contain only a single spatiotemporal frequency component. We used five different combinations of frequencies linearly sampled from $(0, \frac{1}{16}]$ (pixel/ frame), four different temporal frequencies linearly sampled from $(0, \frac{1}{8}]$ (pixel/ frame), and eight different orientations sampled uniformly between $(0, 2\pi]$. In total, there are 160 scenes, and each contains 32 continuous frames. Due to the aperture problem, there is no single solution for this class of motion, but we defined the labels with the slowest moving speed as perpendicular to the spatial stripe, which is most similar to the human interpretation. Incorporating this class of datasets into the training provides a potential slow-world Bayesian prior of humans[10] to solve the ambiguous motion. Check here for a demo of the training sample.

1.2 Training

Environment: For the model training, we implemented our method using the PyTorch 2.0 framework on a workstation equipped with four parallel Nvidia RTX A6000 GPUs, running on the CUDA 11.7 runtime. The data analysis was conducted using MATLAB, as well as Python packages such as SciPy and Pandas.

Timing Setting: Considering the current mainstream playback frame rate (25 FPS) and the time of the human visual impulse response (about 200 ms), we set the temporal window of the first stage to 6 frames (spanning 200 ms). In training, 11 consecutive images (400 ms) are fed into the model. The instantaneous velocity in the middle of the image sequence, i.e., the fifth frame, is used as the label for supervised training.

Hyperparameters: The total trainable parameters of the model is 14.7 M. During training, we set the iteration as eight in stage II. A total of nine flow fields are generated per inference, and we employ a sequence loss based on mean square error [8] to measure the difference between the predicted optical flow and the GT flow. The model is first pre-trained with simple motion and subsequently fine-tuned on complex natural scenes to facilitate convergence[11]. Specifically, we first train the model on simple no-texture motion and drifting grating for 150 epochs with a learning rate (lr) of $1e^{-4}$, and then fine-tune it on the Sintel and DAVIS dataset for an extra 150 epochs with a learning rate of $0.5e^{-4}$. The entire training process was optimized using the Adam optimizer [12] with linear decay of the learning rate and a batch size of 24. Mixed precision training and gradient clipping techniques were employed in all training stages for faster convergence. For image preprocessing, all images were resized to 448×960 and then randomly cropped into 384×576. We also adopted random image horizontal flipping for data augmentation.

2 In Silico Neurophysiological Test

We utilized drifting Gabor or plaid (superimposed by two Gabor components) with a single frequency component as the input stimulus (check here for video demos, Gabor: here; Plaid: here).

Referring to Figure 1 (D), the model's response value at Stage I, or after the remap operation of each iteration in Stage II, was considered as the poststimulus time histogram of the unit. We averaged the responses across spatial dimensions using a Gaussian weighted window to obtain the activation distribution of the total of 256 units, i.e., $\mathbb{R}^{1\times1\times256}$, with respect to input stimulus. The image sequence is generally set to 512×512 with full contrast.

2.1 Directional tuning

We used a single frequency drifting-Gabor and plaid (superimposed by ±30 degrees) as the stimulus input. First, we selected 12 directions uniformly sampled from $(0, 2\pi]$. For each direction, we linearly sampled $8 \times 8 = 64$ sets of spatiotemporal frequency combinations and thus obtained 64 directional tuning curves for each unit using the drifting-Gabor stimulus. We selected the spatiotemporal frequency with the largest tuning standard deviation as the preferred frequency st^* of each unit. Then, we input the Gabor and plaid with the frequency configuration of st^* into the model and obtained the direction tuning curve of each unit. Denoting the model's tuning curve on the drifting-Gabor with st^* as $\mathcal{C}$ and the tuning curve on the plaid with st^* as $\mathcal{P}$, we calculated the directional tuning capability of each cell using a pair of partial correlations [13]:

$$R_{\text{pattern}} = \frac{(r_p - r_c r_{cp})}{\sqrt{\left(\left(1 - r_c^2\right)\left(1 - r_{cp}^2\right)\right)}},\ R_{\text{component}} = \frac{(r_c - r_p r_{cp})}{\sqrt{\left(\left(1 - r_p^2\right)\left(1 - r_{cp}^2\right)\right)}}, \tag{2}$$

where r_c is the correlation of the $\mathcal{P}$ with the component prediction, which is generated by the superimposed ± 30-degree shift of the $\mathcal{C}$; r_p is the correlation of the $\mathcal{P}$ with the pattern prediction, which is equal to $\mathcal{C}$; and r_{cp} is the correlation between the two predictions. We labeled units as "component" if the component correlation coefficient significantly exceeded either 0 or the pattern correlation coefficient, whichever was larger. Similarly, we labeled units as "pattern" if the pattern correlation coefficient significantly exceeded either 0 or the component correlation coefficient. Units were labeled as "unclassified" if both pattern and component correlations significantly exceeded 0 but did not differ significantly from one another, or if neither correlation coefficient differed significantly from 0. The visualized result is plotted on Figure 3 (A) of the main text.

We employed maximum activation technology [14] to render the unit's preferred motion stimuli reversely. The initialized motion stimuli are composed of spatiotemporally incoherent Gaussian white noise $I \in \mathbb{R}^{H \times W \times T}$, which is continuously updated by a gradient ascent method to maximize the activation response of each unit. The activation response of each unit is obtained by a weighted average pooling of its responses $a \in \mathbb{R}^{H \times W}$ across all spatial locations. To consider the continuity of the motion, we add temporal smoothing and spatial smoothing regular terms based on the first-order gradient to the optimization function, which is defined as:

$$\mathcal{A}(I) = \sum_{\Omega} \mathcal{G}_i a_i + w \left(\left| \frac{\partial I(x,y,t)}{\partial x_1} \right| + \left| \frac{\partial I(x,y,t)}{\partial y} \right| + \left| \frac{\partial I(x,y,t)}{\partial T} \right| \right); \tag{3}$$

where Ω denotes the set across the image space and $\mathcal{G}$ is a 2D Gaussian kernel defined on the Ω. We iterate $I := I + w\frac{\partial}{\partial I}\mathcal{A}(I)$ so as to maximize the unit's activation. Each unit went through approximately 300 iterations to generate the following stimuli:

1. For component units, which prefers motion components of a single spatiotemporal frequency: check here for a video demo.
2. For pattern units, which prefers textures or plaids that move in a specific direction: check here for a video demo.
3. For unclassified units, which have a variety of complex motion formats that go beyond the interpretation of simple directional tuning: check here for a video demo.

2.2 Spectral receptive field and speed tuning

To investigate the spectral properties of the units, we used a combination of drifting-Gabor stimuli with different spatiotemporal frequency components. We first sampled 16 directions uniformly from the interval $(0, 2\pi]$. For each direction, we tested $10 \times 10 = 100$ spatiotemporal frequency combinations, with 10 logarithmically spaced spatial frequencies between 0.004 and 0.125 pixels/cycle and 10 logarithmically spaced temporal frequencies between 0.04 and 0.25 pixels/frame. Then we selected the direction with the largest sum of activations across all 100 spatiotemporal combinations as the preferred motion direction $\mathcal{D}^*$. Each unit under its $\mathcal{D}^*$ has 10×10 responses under different spatiotemporal combinations, constituting a grid in logarithmic coordinates. This grid of responses is visualized on the bottom left side of Figure 3 (D) and is referred to as the spectral receptive field of each unit.

Following [15], we use a 2D oriented Gaussian profile to fit the spectral receptive field of each unit, where the 2D Gaussian is defined as:

$$\mathcal{P} = A \cdot \exp\left(\frac{\hat{u}^2}{\sigma_x^2} + \frac{\hat{\omega}^2}{\sigma_y^2}\right) + p,\ s.t. \left\{ \begin{array}{l} \hat{u} = (u - x)\cos\theta + (\omega - y)\sin\theta \\ \hat{\omega} = -(u - x)\sin\theta + (\omega - y)\cos\theta \end{array} \right. \tag{4}$$

The matlab `fmincon` function is used to search the best fitting based on the LSE loss function within $\{(\theta, \sigma_x, \sigma_y, x, y) \mid \theta \in [0, \frac{\pi}{2}); \sigma_{x,y} \in (0, \infty);\ x, y \in [-1, 1]\}$:

$$\ell_{lse} = ||\mathcal{P} - RF|| + w \frac{\sin(2\theta)}{\exp(\frac{|\sigma_1 - \sigma_2|}{\sqrt{\sigma_1 \times \sigma_2}})}, \tag{5}$$

where the second term is a weighted regularization to prevent tilting (i.e., $0 < \theta < 90$) when the receptive field is close to a circle. The fitting results are illustrated in Figure 3 (D) of the main text, where each red dot represents the midpoint of the receptive field and the slope of the bar indicates the orientation of the receptive field, which is consistent with the visualization style in [15].

We further employed a pair of partial correlations to validate the speed-tuning capability for each unit [16]:

$$R_{\text{indep}} = \frac{(r_{\text{i}} - r_{\text{s}} r_{\text{is}})}{\sqrt{(1 - r_{\text{s}}^2)\,(1 - r_{\text{is}}^2)}},\ R_{\text{speed}} = \frac{(r_{\text{s}} - r_{\text{i}} r_{\text{is}})}{\sqrt{(1 - r_{\text{i}}^2)\,(1 - r_{\text{is}}^2)}} \tag{6}$$

where R_{indep} and R_{speed} are the partial correlations of the response field with the independent and speed-tuned predictions, r_i is the correlation of the data with the independent prediction, r_s is the correlation of the data with the speed-tuned prediction, and r_{is} is the correlation of the two predictions, as shown in Figure 3 (F). Here, the spatiotemporal-frequency-independent prediction is computed by taking the outer product of the two 1D tuning curves, and the speed-tuned prediction is computed by shifting the temporal frequency as a function of spatial frequency so that the preferred speed is independent of the speed tuning of the unit. (See [16]'s Figure 5 for details).

3 Psychophysical stimulus Test

Global Motion Integration: The stimulus contained small Gabor blocks with different directions and velocities of motion, which were locally processed by the human V1 cortex and globally integrated by MT to perceive downward motion [17]. Our model exhibited a similar global response to that of humans, as shown in Fig. 4 (A). For the stimulus, please refer to this link.

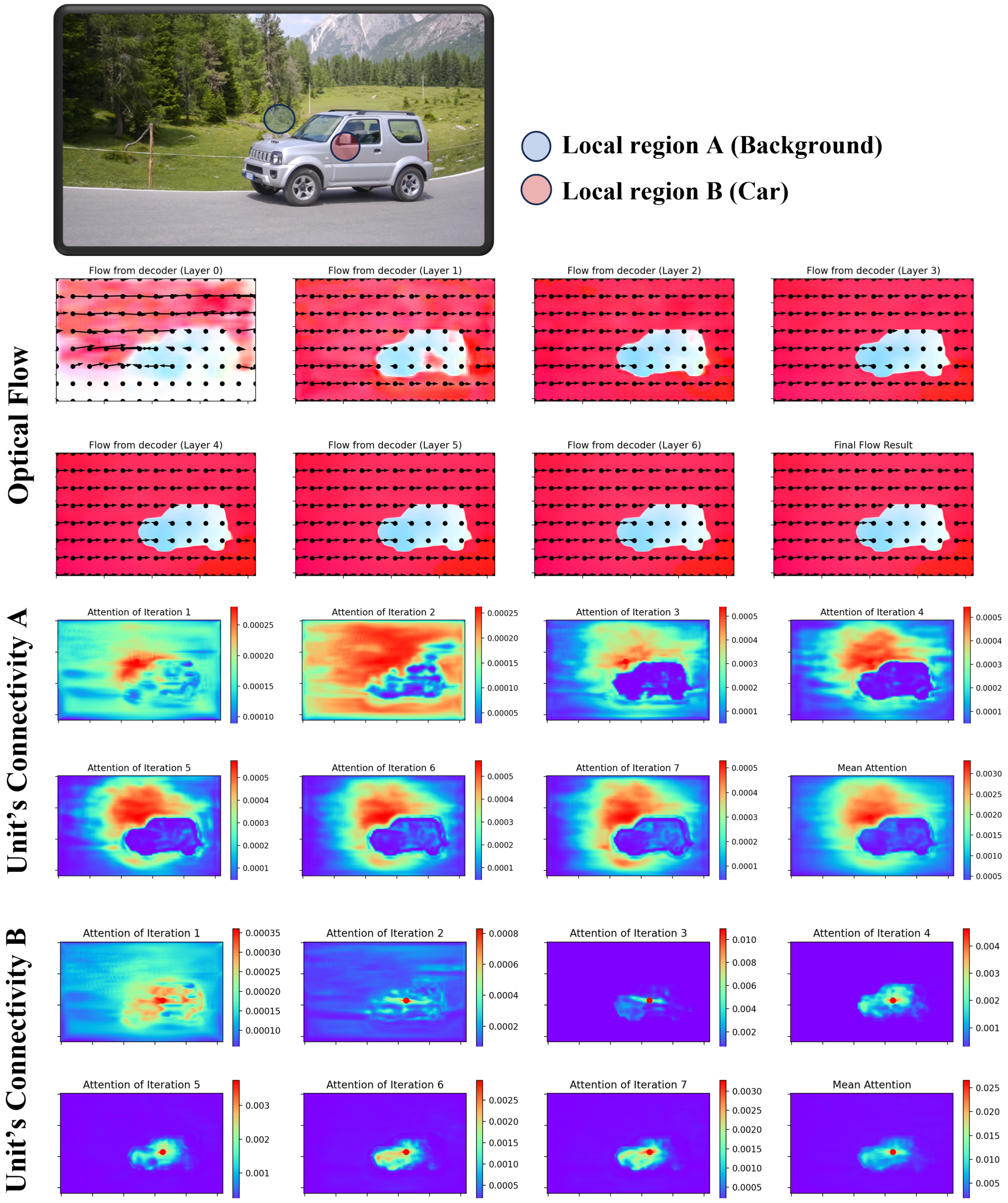
Motion Integration Prototype: As shown in Fig. 4 (B), we presented two stimuli (A: Local-Gabor and B: Local-plaid). For the subjects, stimulus A was more likely to integrate into global downward motion than stimulus B [17]. This indicates that humans preferentially integrate motion in conjunction with local cues, and once the aperture problem is solved locally (e.g., stimulus B), the ability of motion to propagate distally will be inhibited. The model demonstrated the same integration strategy as humans. Please refer to this link for stimulus A and this link for stimulus B.

Missing-Fundamental Motion: Check here for a detailed view. This stimulus induced a strong illusion of leftward motion by adding a leftward Fourier motion to the rightward moving square wave and removing the first harmonic of its Fourier series [18]. Visually, the subject perceives movement to the left[18], but if focusing on structure alone, the stimulus is moving to the right (check here to experience). Most DNN models fail to handle this motion stably or perceive rightward motion, whereas our model perceives leftward Fourier motion consistent with human perception. Check here for the result of RAFT[8] , here for the result of GMFlow[19].

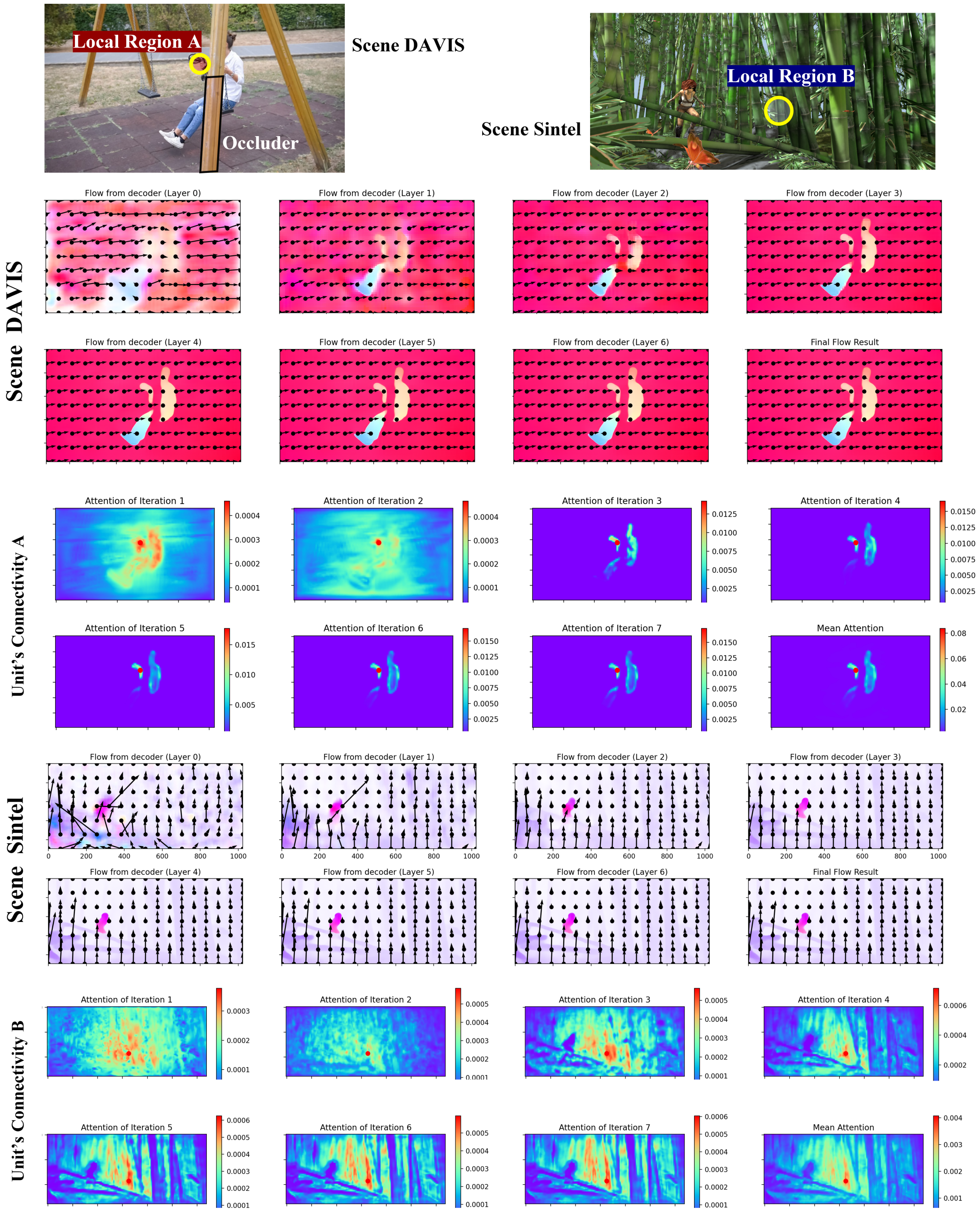
Mario Reversed-Phi Illusion[20]: For the video demo, please refer to this link. This stimulus demonstrated the reverse-phi illusion in psychology, where the subject perceives continuous motion despite the absence of actual motion. Current DNN models provide unstable results for this kind of motion (see [20], also click here for the result of FlowNet2.0[11]), while our model produces the illusion of motion similar to the continuous motion perceived by humans; check this link for our result.

Results on Naturalistic Scenes: The proposed motion integration mechanism demonstrated sufficient flexibility to cope with various complex natural scenes and is comparable to human and current DNN models. Without specific fine-tuning, our best results had an end-point error of 1.69 on the clean of the Sintel training set and 1.75 on the final set, which is comparable to the majority of DNN models. For a demo on the Sintel dataset, please refer to this link.

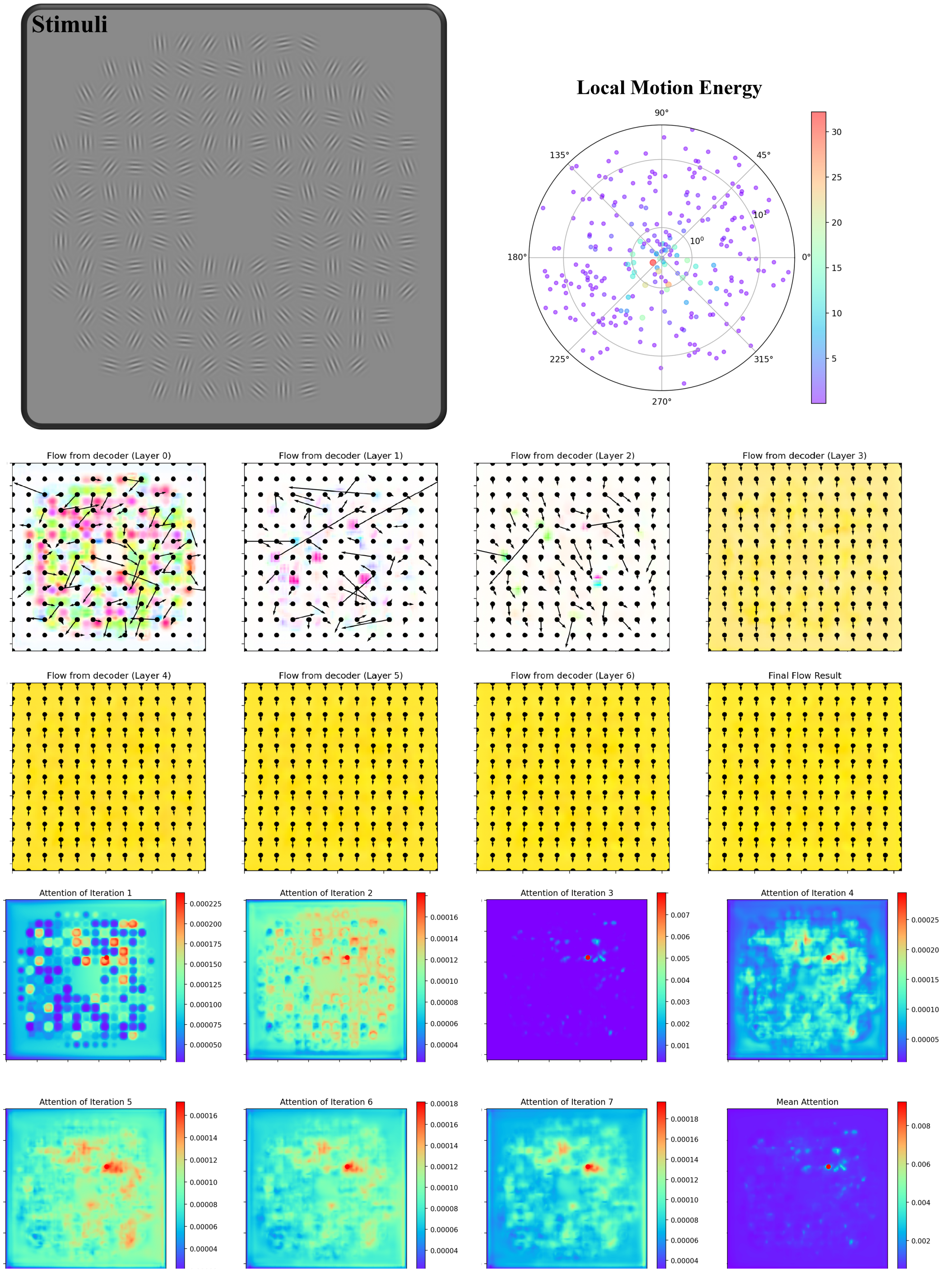
The following is demonstration diagrams of the motion integration and segregation process:



Appendix-Fig. 1: ***Demo of motion integration process.*** Foreground and background can be segmented by leveraging the differences in connectivity within unit clusters, similar to segmentation tasks.



Appendix-Fig. 2: ***Demo of adaptive motion integration & segregation process.*** The example shows that our model can handle intricate natural scenes and consider certain occlusion relations, where the motion energy in region A demonstrates its robust connectivity within the graph (attention) space, even when different body regions are separated due to occlusion.



Appendix-Fig. 3: ***Demo of global motion integration.*** Our model successfully replicated the illusion of perceiving a global downward motion from Stage I to Stage II, similar to the integration of motion signals from V1 cells in the MT region of humans.

References

[1] R. Jonschkowski, A. Stone, J. T. Barron, A. Gordon, K. Konolige, and A. Angelova, “What matters in unsupervised optical flow,” in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*, pp. 557–572, Springer, 2020.

[2] A. Stone, D. Maurer, A. Ayvaci, A. Angelova, and R. Jonschkowski, “Smurf: Self-teaching multi-frame unsupervised raft with full-image warping,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3887–3896, 2021.

[3] S. Meister, J. Hur, and S. Roth, “Unflow: Unsupervised learning of optical flow with a bidirectional census loss,” in *Proceedings of the AAAI conference on artificial intelligence*, vol. 32, 2018.

[4] Y.-H. Yang, T. Fukiage, Z. Sun, and S. Nishida, “Psychophysical measurement of perceived motion flow of naturalistic scenes,” *bioRxiv*, 2023.

[5] D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black, “A naturalistic open source movie for optical flow evaluation,” in *European Conf. on Computer Vision (ECCV)* (A. Fitzgibbon et al. (Eds.), ed.), Part IV, LNCS 7577, pp. 611–625, Springer-Verlag, Oct. 2012.

[6] F. Perazzi, J. Pont-Tuset, B. McWilliams, L. Van Gool, M. Gross, and A. Sorkine-Hornung, “A benchmark dataset and evaluation methodology for video object segmentation,” in *Computer Vision and Pattern Recognition*, 2016.

[7] Z. Huang, X. Shi, C. Zhang, Q. Wang, K. C. Cheung, H. Qin, J. Dai, and H. Li, “Flowformer: A transformer architecture for optical flow,” in *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XVII*, pp. 668–685, Springer, 2022.

[8] Z. Teed and J. Deng, “Raft: Recurrent all-pairs field transforms for optical flow,” in *European conference on computer vision*, pp. 402–419, Springer, 2020.

[9] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, “Slic superpixels compared to state-of-the-art superpixel methods,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 34, no. 11, pp. 2274–2282, 2012.

[10] Y. Weiss, E. P. Simoncelli, and E. H. Adelson, “Motion illusions as optimal percepts,” *Nature neuroscience*, vol. 5, no. 6, pp. 598–604, 2002.

[11] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, and T. Brox, “Flownet 2.0: Evolution of optical flow estimation with deep networks,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2462–2470, 2017.

[12] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *arXiv preprint arXiv:1412.6980*, 2014.

[13] C. C.-R. G.-C. GROSS, “Pattern recognition mechanisms,” 1983.

[14] J. Yosinski, J. Clune, A. Nguyen, T. Fuchs, and H. Lipson, “Understanding neural networks through deep visualization,” *arXiv preprint arXiv:1506.06579*, 2015.

[15] J. A. Perrone and A. Thiele, “Speed skills: measuring the visual speed analyzing properties of primate mt neurons,” *Nature neuroscience*, vol. 4, no. 5, pp. 526–532, 2001.

[16] N. J. Priebe, C. R. Cassanello, and S. G. Lisberger, “The neural representation of speed in macaque area mt/v5,” *Journal of Neuroscience*, vol. 23, no. 13, pp. 5650–5661, 2003.

[17] K. Amano, M. Edwards, D. R. Badcock, and S. Nishida, “Adaptive pooling of visual motion signals by the human visual system revealed with a novel multi-element stimulus,” *Journal of vision*, vol. 9, no. 3, pp. 4–4, 2009.

[18] R. O. Brown and S. He, “Visual motion of missing-fundamental patterns: motion energy versus feature correspondence,” *Vision Research*, vol. 40, no. 16, pp. 2135–2147, 2000.

[19] H. Xu, J. Zhang, J. Cai, H. Rezatofighi, and D. Tao, “Gmflow: Learning optical flow via global matching,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8121–8130, 2022.

[20] J. Yates, “Motion illusions.” `https://jake.vision/blog/motion-illusions`, Dec 2020.