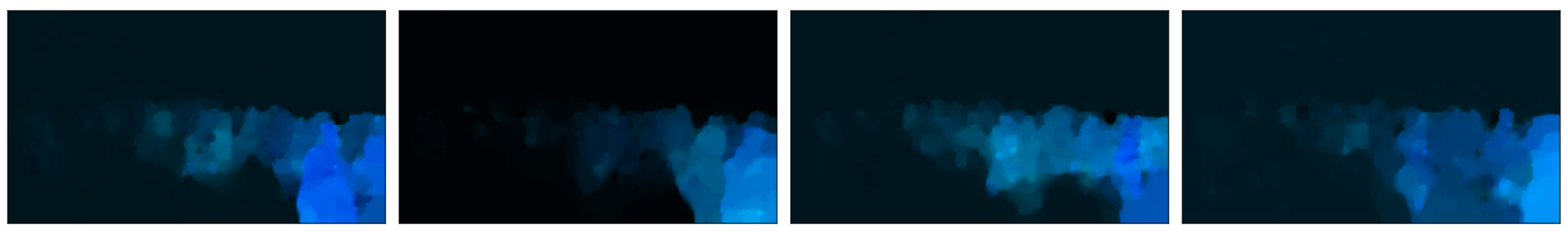
Flow Dynamics Correction for Action Recognition

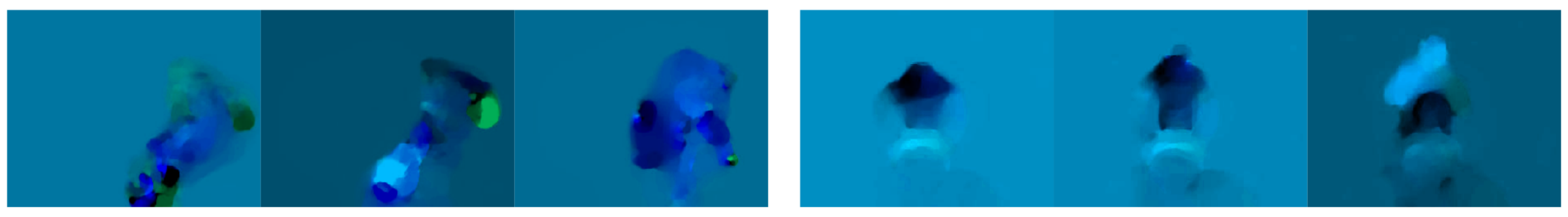
Lei.W@anu.edu.au[1,2]   Piotr.Koniusz@data61.csiro.au[2,1]

[1]Australian National University [2]Data61/CSIRO

Motivation and key ideas



(a) *Marathon*: stride =1, 4, 8 and 12 respectively (from left to right).



(b) *Kick ball*: stride =1, 2 and 4. (c) *Situp*: stride =1, 2 and 4.

- **Multi-stride optical flow** (LDOF) on (a) YUP++ & (b) – (c) HMDB-51. Different strides (temporal scales) capture different granularity levels of motions. The visual appearance varies between different strides.
- **Optical flow correction.** Let $\mathbf{U}$ and $\mathbf{V}$ be two maps with the displacement components (along x and y axis, respectively) of the computed multi-stride optical flow. The magnitude and angle of the optical flow $(\mathbf{U}, \mathbf{V})$ are computed (by element-wise operations) as

$$\boldsymbol{M} = \sqrt{\mathbf{U}^2 + \mathbf{V}^2}, \qquad (1)$$

$$\boldsymbol{\Phi} = \arctan(\mathbf{U}/\mathbf{V}). \qquad (2)$$

As videos are highly affected by many issues like noise, camera shaking, dynamic background environments and a mixture of fast and slow motions, *e.g.*, human actions, we apply the element-wise power normalization (PN), on the magnitude component $\boldsymbol{M}$ for the flow correction to get the power normalized magnitude matrix $\boldsymbol{M}'$

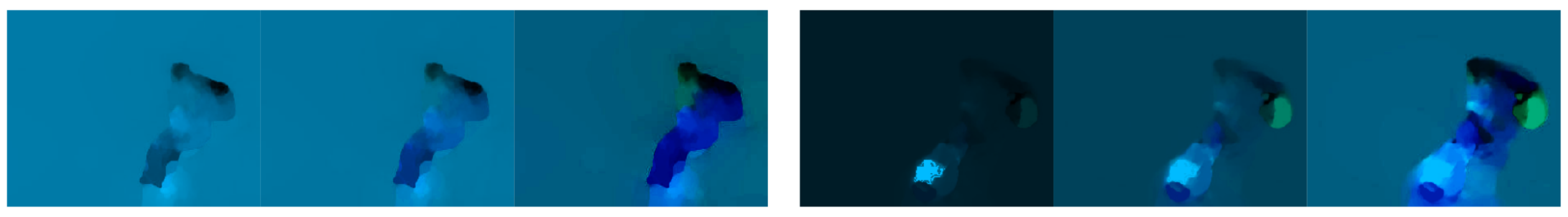
$$\boldsymbol{M}' = \operatorname{sign}(\boldsymbol{M}) \cdot (1 - (1 - \operatorname{abs}(\boldsymbol{M}))^{\gamma}), \qquad (3)$$

where $\gamma > 0$ decides the strength of PN, and all operations are element-wise. The PN here is used for the flow correction that is performed on each optical flow frame. The normalization is done on the magnitude component of the optical flow so as to boost or dampen subtle or sudden motions. We then compute optical flow features (OFFs) from such mended motion clips. The use of abs and sign in Eq. (3) is for maintaining the motion direction.

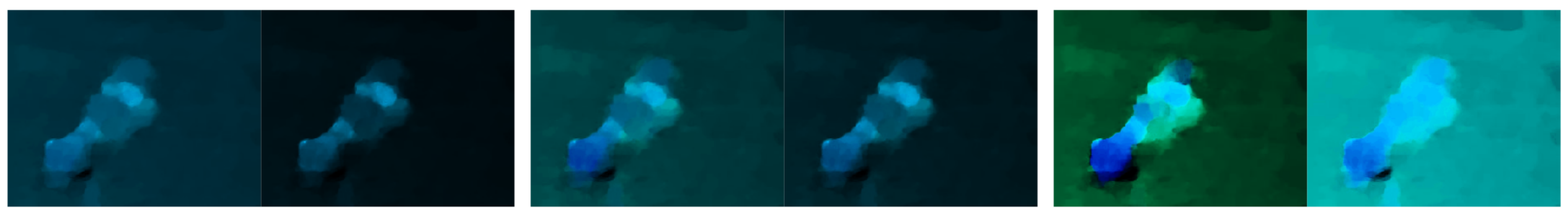
- We use $\gamma > 1$ to boost weak and dampen dominant motions (*c.f.* $0 < \gamma < 1$ to preserve only dominant motions) which gives us selective focus on various motion dynamics. Note that if $\gamma = 1$, PN is not performed. Finally, we recover two optical flow maps $(\mathbf{U}', \mathbf{V}')$ based on the corrected $\boldsymbol{M}'$ and $\boldsymbol{\Phi}$ as

$$\mathbf{U}' = \boldsymbol{M}' \cdot \sin(\boldsymbol{\Phi}), \qquad (4)$$

$$\mathbf{V}' = \boldsymbol{M}' \cdot \cos(\boldsymbol{\Phi}). \qquad (5)$$

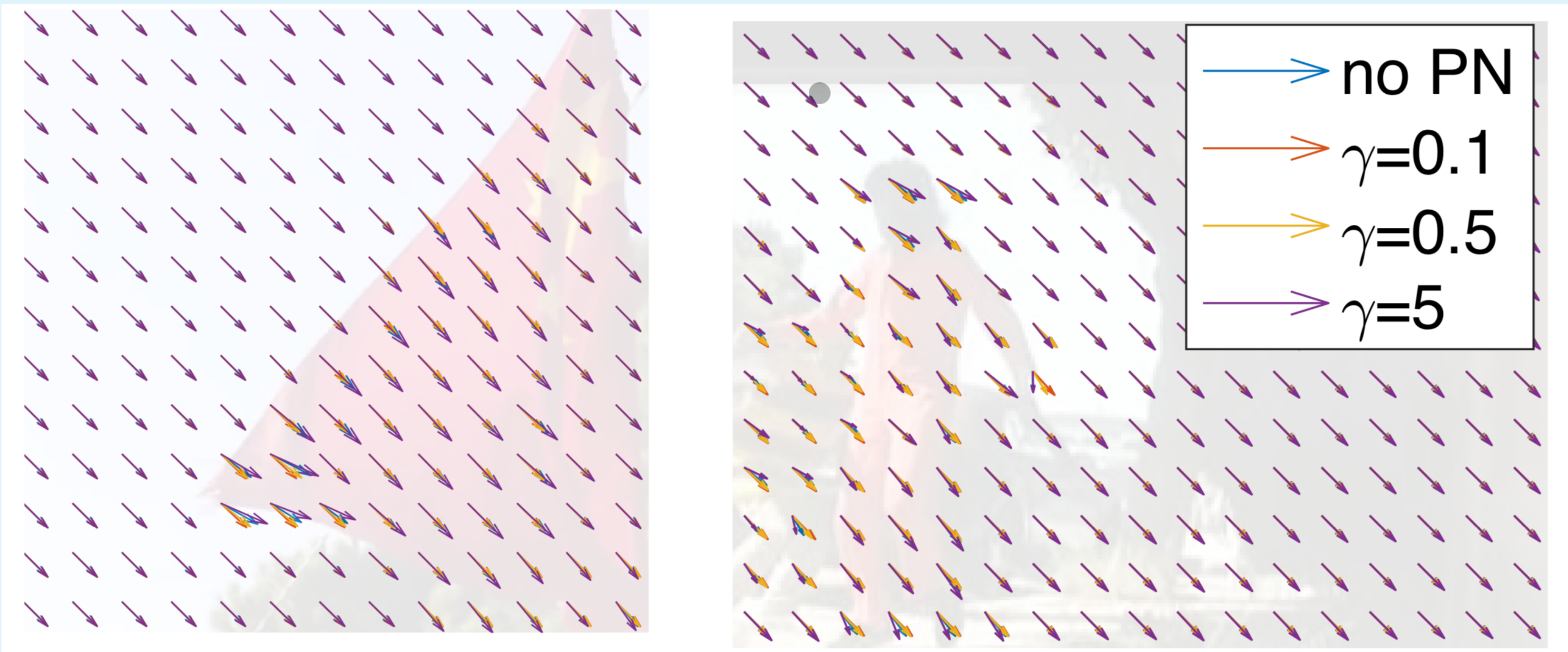


(a) DeepFlow: γ =0.1, 0.5 and 5. (b) TV-L1: γ =0.1, 0.5 and 5.

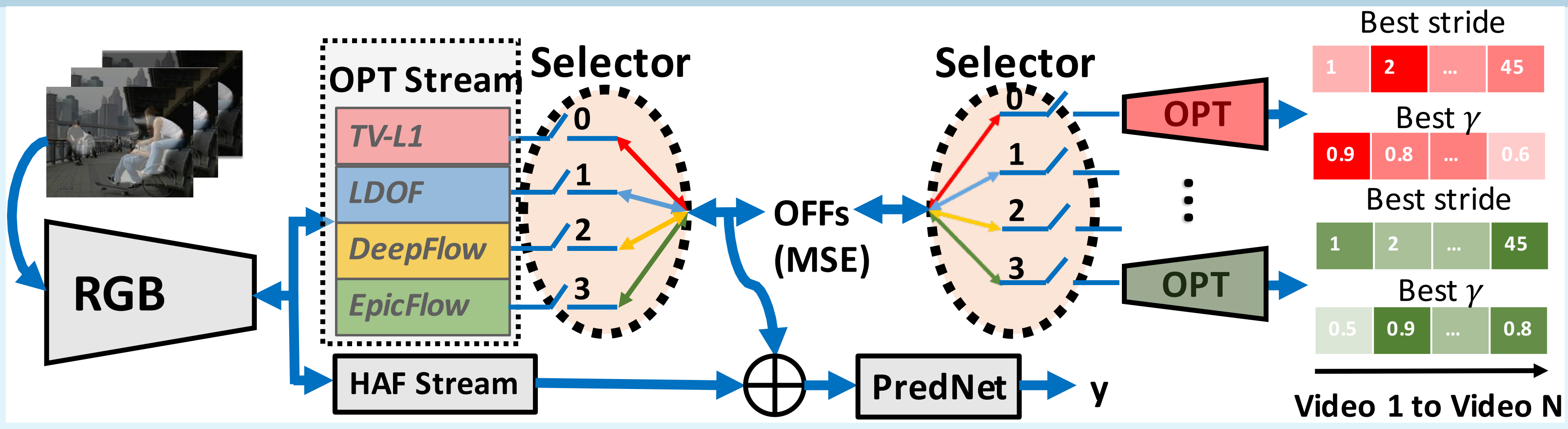


(c) EpicFlow: $\gamma = 0.1$. (d) EpicFlow: $\gamma = 0.5$. (e) EpicFlow: $\gamma = 5$.

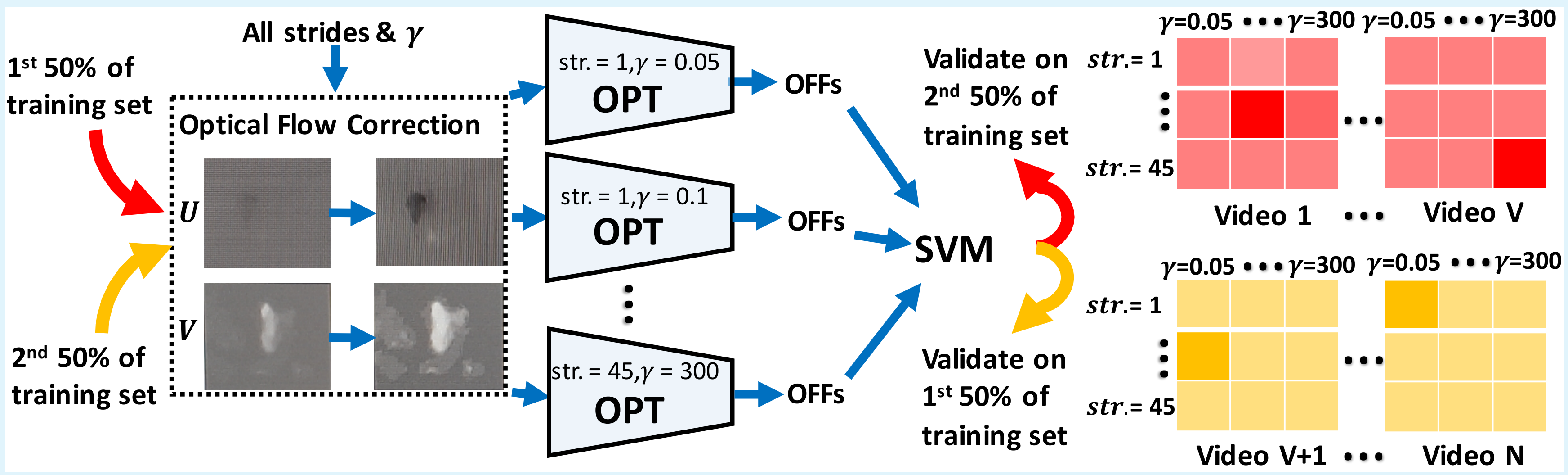
(a)–(b) show the strength of PN (γ) for optical flow correction on action *Kick ball*. Small γ preserves the dominant motions and large γ boosts some weak motions and maintains more rich motion dynamics. Each pair of figures in (c) – (e) shows with (left) and without (right) dominant motions on action *dribble*. All actions are from HMDB-51.



The pipeline: further details

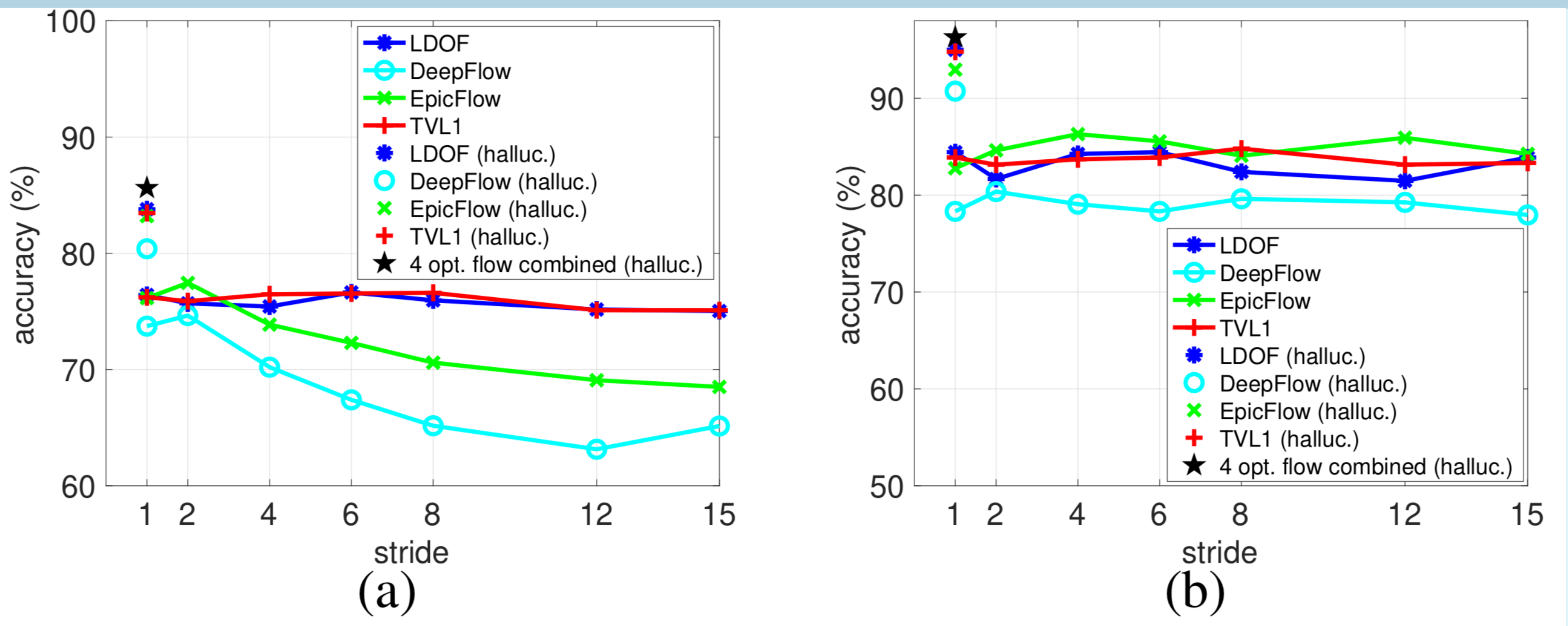


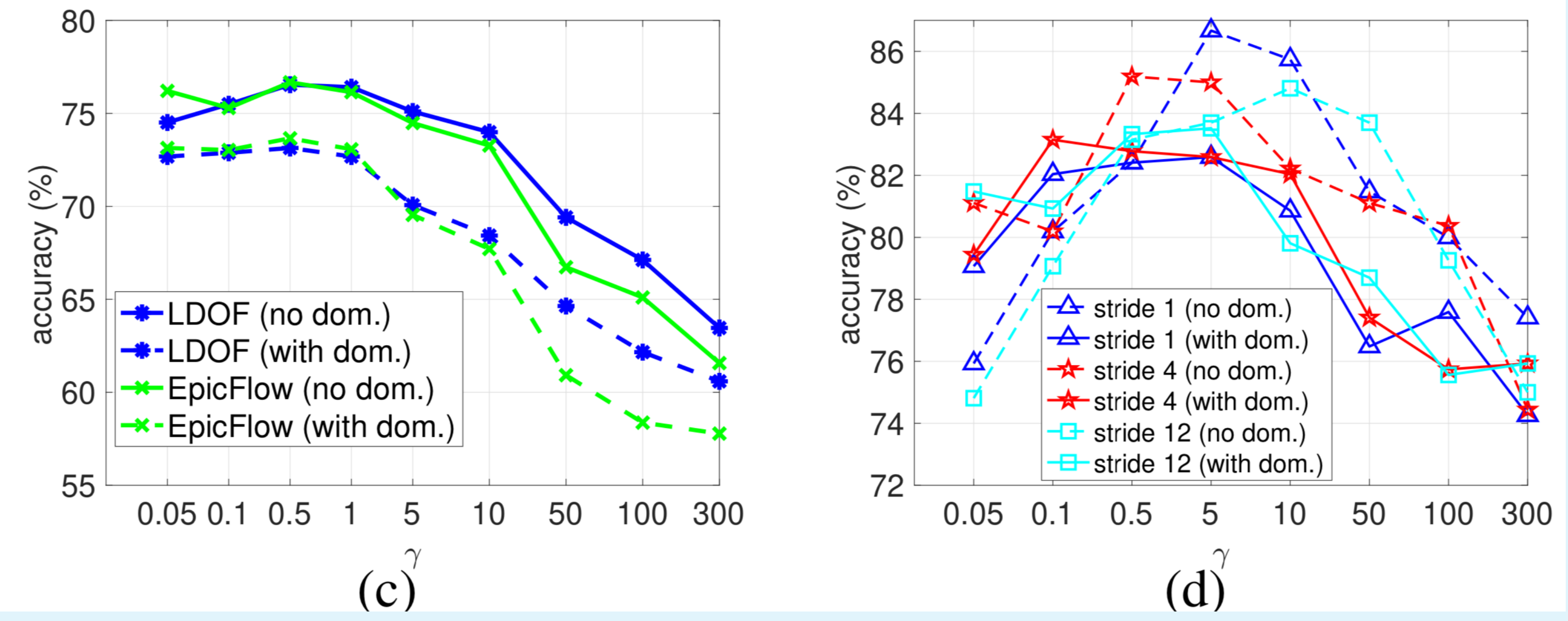
- We use optical flow (OPT) streams and a **Selector** to learn to hallucinate the best optical flow features (OFFs). The OFFs and features from the High Abstraction Features (HAF) stream are concatenated by $\bigoplus$, and then feed into the PredNet (a simple MLP) for classification.
- The Selector is used to choose the optical flow types we learn to hallucinate, given the best OFFs. MSE represents the mean square error loss, while y denotes the output class label from PredNet.
- The input to our HAL is the RGB video, and it learns (during training) to translate latent features from RGB into OFFs, which represent various motion dynamics based on optical flow. Our pipeline uses the corrected flow dynamics. There are 4 switches that activate the corresponding optical flow streams based on the selection.



- Given the corrected optical flow, we split train data into two halves.
- We train scoring optical flow networks (*e.g.*, I3D or AssembleNet/AssembleNet++ optical flow stream pre-trained on Kinetics-400), one per optical flow type, stride choice and γ choice.
- We train on one half of train data, and score via SVM each video on the second half of train data in terms of which (stride, γ) recognises video correctly (or is the closest to correct decision).
- We train networks on the second half of the train data and score videos on the first half. With such scoring, we can train four optical flow networks by directing to them best (stride, γ) per video.

Results





	I3D	DEEP-HAL	AssembleNet	AssembleNet++	HAL (ours, (with I3D)	HAL (ours, AssembleNet++)
Original	40.0	43.1	56.6	59.8	45.3	**62.0**
Flow Corr. (ours)	**42.1**	**45.7**	**59.7**	**62.0**	**48.7**	**64.9**
Improvement	↑**2.1**	↑**2.6**	↑**3.1**	↑**2.2**	↑**3.4**	↑**2.9**

ActionCLIP 44.3 SlowFast 45.2 En-VidTr-L 47.3 MoViNet-A6 63.2 TubeViT-L 66.2

Table 1: Evaluations of various methods (*top*) w/wo flow dynamics correction and (*bottom*) comparisons to the state of the art on Charades.