Shift Equivariant Pose Network (Supplementary Materials)

Anonymous WACV Algorithms Track submission

Paper ID 985

A. Introduction to APS

APS-D. [3] APS is proposed to make symmetric downsampling and upsampling operations shift equivariant. Pipeline of adaptive polyphase downsampling (APS-D) is shown in Figure 1. The input image can be grouped as 4 possible grids, each is a potential candidate for downsampling. These 4 possible grids are called *polyphase components* and denoted as $\{y_{ij}\}_{i,j=0}^{1}$. Adaptive polyphase downsampling (APS-D) then can be formulated as

$$\begin{aligned} y_{APS-D} &= y_{i_1 j_1}, \\ where \quad i_1, j_1 &= \arg\max_{i,j}\{||y_{ij}||_p\}_{i,j=0}^{1}. \end{aligned} \tag{1}$$

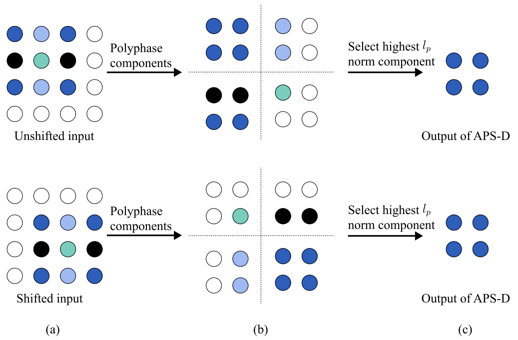

Figure 1. Pipeline of APS-D on single channel input. (a) Input and its shift. (b) The input and its shift share the same set of polyphase components. (c) By choosing the component with the highest lp norm, APS-D returns the same output for both the input and its shift.

APS-U. [2] Adaptive polyphase upsampling (APS-U) is proposed to recover the shift-consistent APS-D output to a shift equivariant one. The sampled polyphase components can be denoted as $\{y_{ij}\}_{i,j=0}^{1}$. We denote $i_x, x \in \{0, 1, 2, 3\}$ to be the index of polyphase component with the highest norm. Then the downsampled output obtained from APS-D is

$$y_{APS-D} = D_2(T_{-i_x}(x)). \tag{2}$$

Let U_2^A denote adaptive polyphase upsampling (APS-U) operator with stride 2. D_2^A denote adaptive polyphase downsampling (APS-D) operator with stride 2. Then the adaptive polyphase upsampling (APS-U) is formulated as

$$U_2^A(y_{APS-D}, i_x) = T_{i_x}(U_2((y_{APS-D})) \tag{3}$$

When input signal is shifted by odd and even pixel, the APS-D and APS-U pipeline is shown in Figure 2.

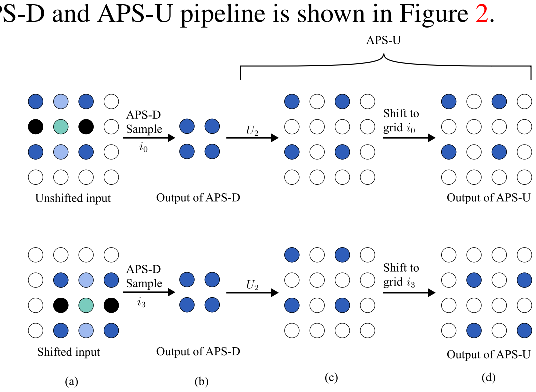

Figure 2. (a)-(b) output of APS-D is consistent to input translations. (c) Outputs of conventional upsampling are not shifted in same direction and stride as the input. (d) By shifting the upsampled signal to the grid originally chosen by APS-D, shift equivariance is achieved.

B. GDPE using Newton-Raphson method

GDPE (**G**aussian **D**istribution prior based keypoint **P**arameter **E**stimation) is a kind of Iterative Parameter Estimation Method. Given a predicted heatmap $\hat{C} = \{(c_0, x_0, y_0), (c_1, x_1, y_1), ..., (c_p, x_p, y_p)\}, p \in \mathbb{Z}$, we aim to fit $\hat{C}$ through the model, so that $\hat{C}$ is getting closer and closer to C_{GT}.

When the model converges, it is reasonable that $\hat{C}$ should follow the same distribution as C_{GT}. Then we are able to estimate the parameter $\mu = (m, n)$ with gaussian distribution prior and pre-known σ. So the whole process of GDPE can be formulated as: solving (m,n) from

$$C(x,y,m,n)=\exp(-\frac{(x-m)^2+(y-n)^2}{2\sigma^2}), \tag{4}$$

given data $\{(c_0,x_0,y_0),(c_1,x_1,y_1),...,(c_p,x_p,y_p),p\in\mathbb{Z}\}$ and σ. We take the logarithm of both sides:

$$-2\sigma^2\ln(C(x,y,m,n))=(x-m)^2+(y-n)^2. \tag{5}$$

Let

$$\mathcal{C}(x,y,m,n)=-2\sigma^2\ln(C(x,y,m,n)), \tag{6}$$

then we have:

$$\mathcal{C}(x,y,m,n)=(x-m)^2+(y-n)^2. \tag{7}$$

Let the objective optimization function be

$$F(m,n)=\sum_{i=0}^{n}[\mathcal{C}_i-(x_i-m)^2-(y_i-n)^2]^2. \tag{8}$$

In order to make $F(m,n)$ reach a minimum value, it is necessary to make:

$$\begin{cases}\dfrac{\partial F}{\partial m}=0\\[2ex] \dfrac{\partial F}{\partial n}=0\end{cases}, \tag{9}$$

which is

$$\begin{cases}\sum_{i=0}^{n}4(x_i-m)(\mathcal{C}_i-(x_i-m)^2-(y_i-n)^2)=0\\ \sum_{i=0}^{n}4(y_i-n)(\mathcal{C}_i-(x_i-m)^2-(y_i-n)^2)=0\end{cases}. \tag{10}$$

Use Newton-Raphson method [1] to solve the nonlinear equations. Assume:

$$\begin{cases}f(m,n)=\sum_{i=0}^{n}4(x_i-m)(\mathcal{C}_i-(x_i-m)^2-(y_i-n)^2)\\ g(m,n)=\sum_{i=0}^{n}4(y_i-n)(\mathcal{C}_i-(x_i-m)^2-(y_i-n)^2)\end{cases}. \tag{11}$$

The first step, given the initial value (m_0,n_0), the approximate equation is

$$\begin{cases}f(m_0,n_0)+\left.\dfrac{\partial f}{\partial m}\right|_{m=m_0,n=n_0}(m-m_0)+\left.\dfrac{\partial f}{\partial n}\right|_{m=m_0,n=n_0}(n-n_0)=0\\[2ex] g(m_0,n_0)+\left.\dfrac{\partial g}{\partial m}\right|_{m=m_0,n=n_0}(m-m_0)+\left.\dfrac{\partial g}{\partial n}\right|_{m=m_0,n=n_0}(n-n_0)=0\end{cases}. \tag{12}$$

The second step, we can solve the above approximate equation:

$$m_1=\frac{(\frac{\partial f}{\partial m}n_0+\frac{\partial f}{\partial n}n_0-f)\frac{\partial g}{\partial n}-(\frac{\partial g}{\partial m}m_0+\frac{\partial g}{\partial n}n_0-g)\frac{\partial f}{\partial n}}{\frac{\partial f}{\partial m}\frac{\partial g}{\partial n}-\frac{\partial f}{\partial n}\frac{\partial g}{\partial m}}, \tag{13}$$

$$n_1=\frac{(\frac{\partial f}{\partial m}n_0+\frac{\partial f}{\partial n}n_0-f)\frac{\partial g}{\partial m}-(\frac{\partial g}{\partial m}m_0+\frac{\partial g}{\partial n}n_0-g)\frac{\partial f}{\partial m}}{\frac{\partial f}{\partial n}\frac{\partial g}{\partial m}-\frac{\partial f}{\partial m}\frac{\partial g}{\partial n}}. \tag{14}$$

The third step, repeat the above operations, and after multiple iterations we can get the approximate solution until:

$$|m_i-m_{i-1}|+|n_i-n_{i-1}|<\epsilon. \tag{15}$$

Then it is considered that (m_i,n_i) at this time is the most accurate approximate solution of the objective function.

Newton-Raphson method has square-convergence properties, which means that with each iteration, the significant number of approximate roots will double. Only after a few iterations, the estimated results will be accurate enough as the ground truth.

C. More visualizations of shift-equivariance

Figure 3 and Figure 4 shows the visualization results of our method when the input is shifted by 0-8 pixel values. We also attach two videos, and recommend the reviewers refer to them for more intuitive comparison.

D. Code

The code for reproducing our model is attached.

References

[1] Saba Akram and Quarrat Ul Ann. Newton raphson method. *International Journal of Scientific & Engineering Research*, 6(7):1748–1752, 2015. 2

[2] Anadi Chaman and Ivan Dokmanić. Truly shift-equivariant convolutional neural networks with adaptive polyphase upsampling. *arXiv preprint arXiv:2105.04040*, 2021. 1

[3] Anadi Chaman and Ivan Dokmanic. Truly shift-invariant convolutional neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3773–3783, 2021. 1

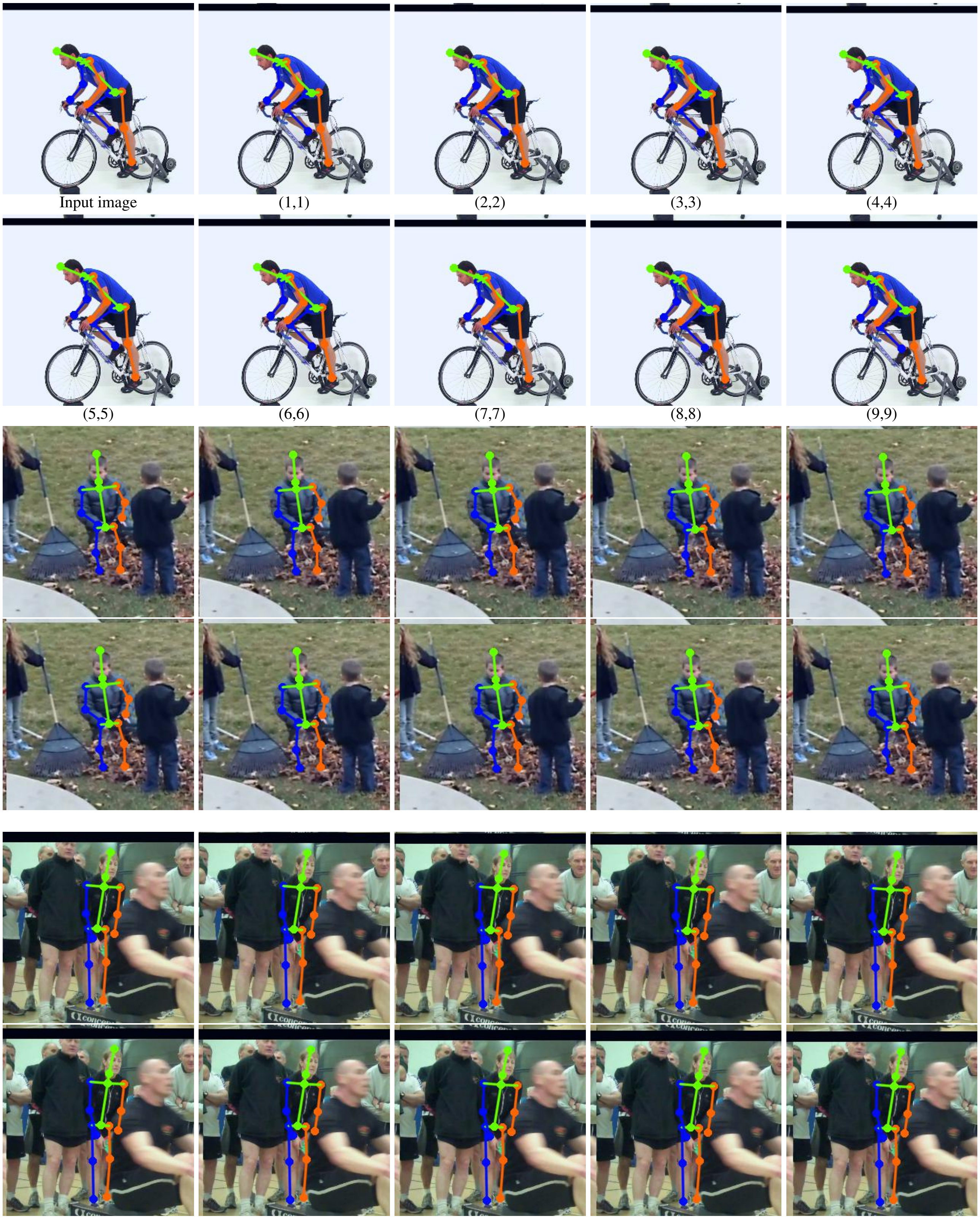

Figure 3. More visualization results of our method when the input is shifted by $(\Delta x, \Delta y)$ pixels values.

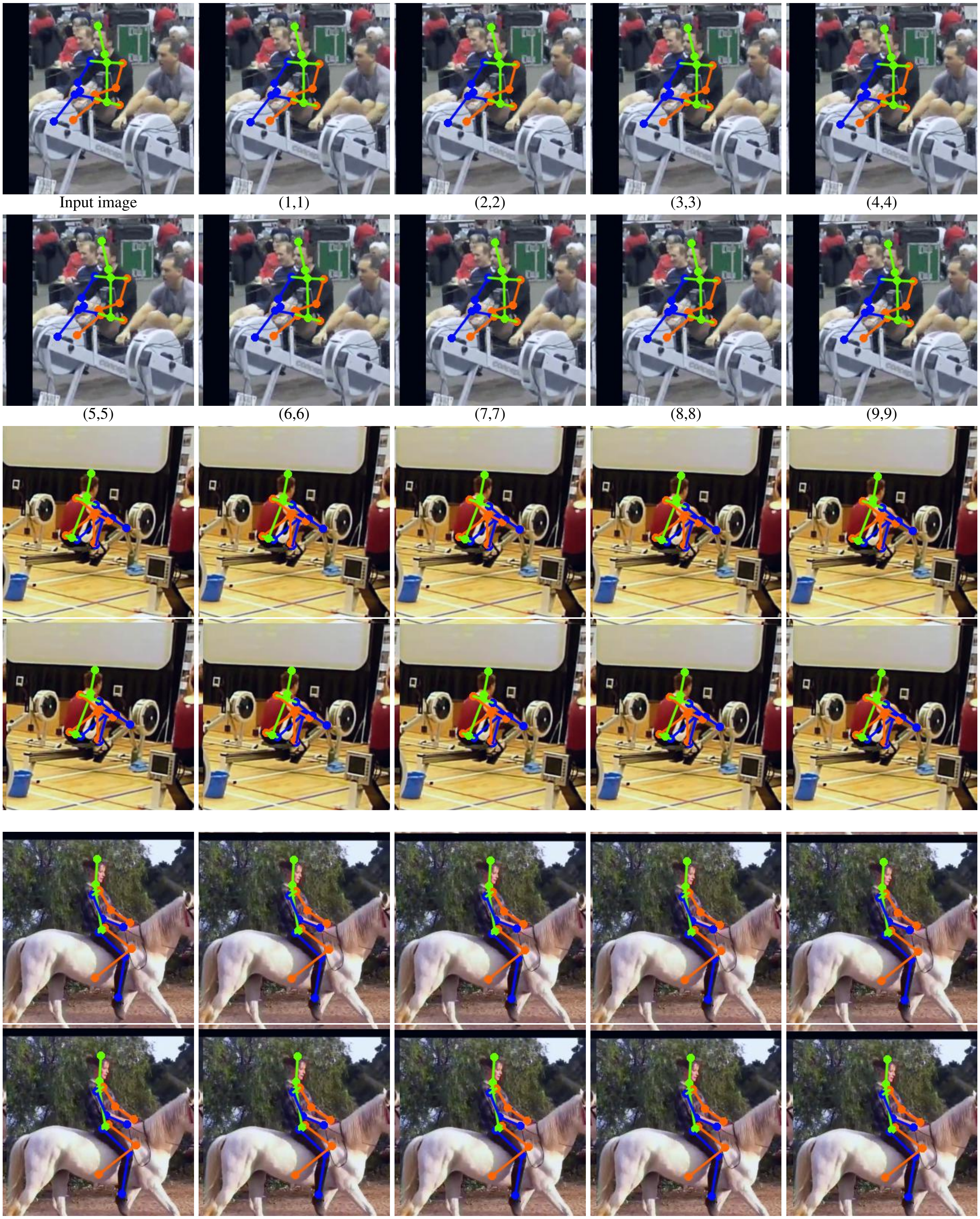

Figure 4. More visualization results of our method when the input is shifted by $(\Delta x, \Delta y)$ pixels values.