Optical flow estimation in spherical images using deep learning approaches. No omnidirectional annotated dataset; Strong images distortions due to projection. Adaptation of perspective networks taking into account distortions by modifying the convolution.

Adapting the perspective CNN kernels to equirectangular: (Fernandez et al. 2020)

\[ \phi_0 = \left( \omega_0 - \frac{W}{2} \right) \frac{2\pi}{W} \quad \theta_0 = -\left( \omega_0 - \frac{H}{2} \right) \frac{\pi}{H} \]  

For a kernel size \( r \) with local points \( (i, j) \) in \( [-\frac{r-1}{2}, \frac{r-1}{2}] \), the coordinates of these points on the local perspective projection are:

\[ p_{\text{persp}, ij} = \begin{bmatrix} x_{ij} \\ y_{ij} \\ z_{ij} \end{bmatrix} = R_\theta (\phi_0) R_\phi (\theta_0) \frac{p_{\text{sph}, ij}}{\|p_{\text{sph}, ij}\|} \]  

where \( R_\theta (\beta) \) rotation matrix of \( \beta \) around the a axis.

Finally, the back projection on the 2D equirectangular image gives these coordinates for the kernel points:

\[ a_{ij} = \frac{W}{2\pi} \left( \arctan \left( \frac{x_{ij}}{y_{ij}} \right) + \pi \right) ; \quad v_{ij} = -\frac{H}{\pi} \left( \arcsin (y_{ij}) - \frac{\pi}{2} \right) . \]  


1. Training

2. Testing

Ground truth optical flow extracted using Blender Vector Pass (Ranjan et al. 2020).

Validation on virtual datasets built on Blender

Validation on real videos

OmniflowNet:

- perspective CNN adapted to equirectangular images;
- plugin transferable on any CAFFE CNN;
- no extra training needed;
- no slowdown in the CNN execution;
- proven performances on real and virtual datasets.

References


OmniflowNet LiteFlowNet2 RGB Input

OmniFlowNet

http://www.i3s.unice.fr/~allibert/Videos/icpr20_video.mp4.