Deformable Convolutions

PA228: Machine Learning in Image Processing

Vasilii Karmazin

Methodology

(a) standard convolution   (b) deformable convolution

Offset field

Conv

2K K

Amplitude field

Deformable convolution
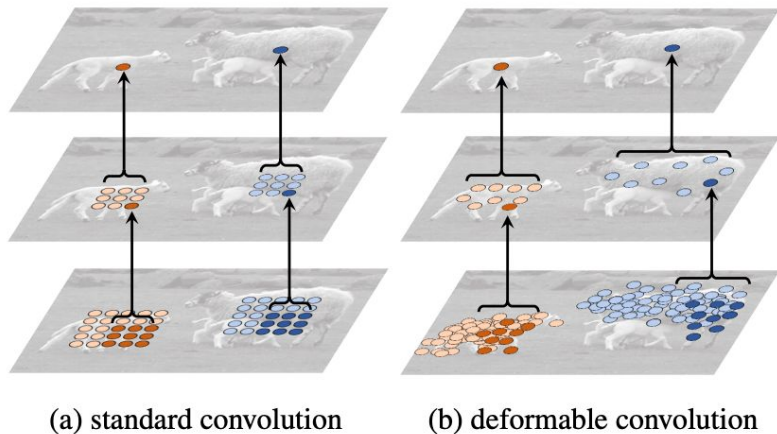
- Can handle geometric transformations in object scale, pose, viewpoint, and part deformation
- The 2D offset vectors are learned as an addition to the regular grid sampling locations
- Modulation mechanism allow network to control signal strength from a particular location. $\Delta m \in [0, 1]$
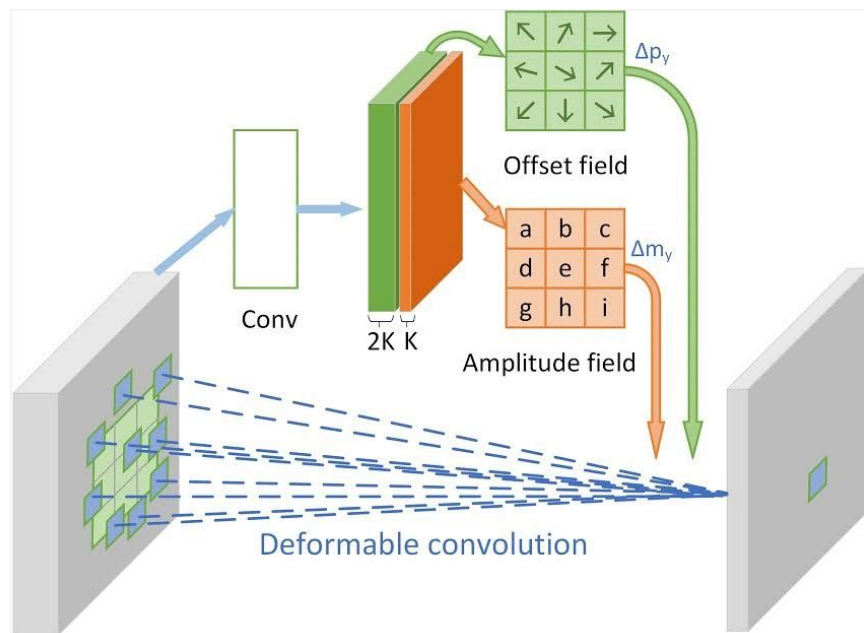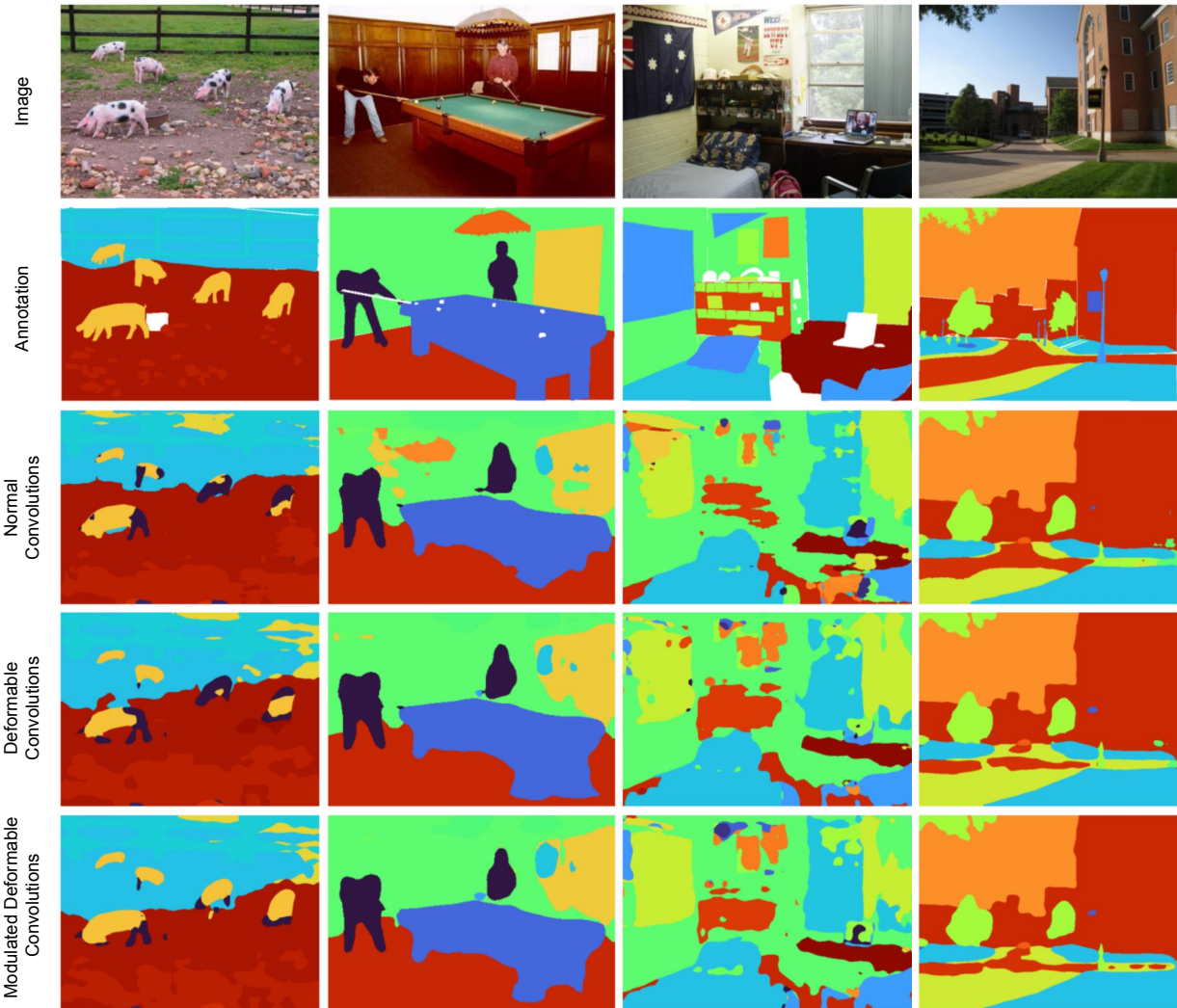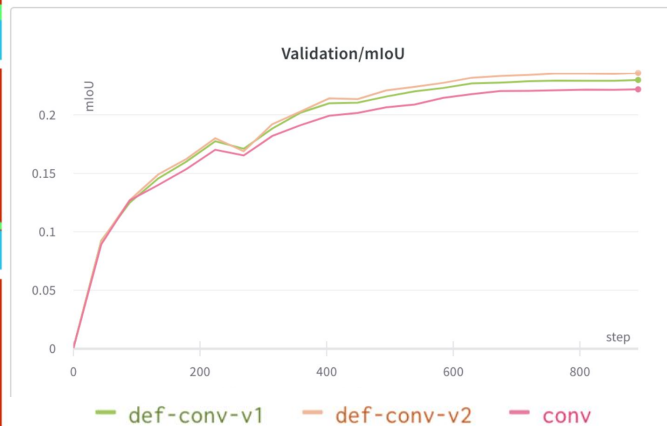
Conv: $\quad \mathbf{y}(\mathbf{p}_0) = \sum_{\mathbf{p}_n \in \mathcal{R}} \mathbf{w}(\mathbf{p}_n) \cdot \mathbf{x}(\mathbf{p}_0 + \mathbf{p}_n)$ $\quad$ (1)

Def Conv: $\quad \mathbf{y}(\mathbf{p}_0) = \sum_{\mathbf{p}_n \in \mathcal{R}} \mathbf{w}(\mathbf{p}_n) \cdot \mathbf{x}(\mathbf{p}_0 + \mathbf{p}_n + \Delta \mathbf{p}_n)$ $\quad$ (2)

Modulated Def Conv: $\quad y(p) = \sum_{k=1}^{K} w_k \cdot x(p + p_k + \Delta p_k) \cdot \Delta m_k$ $\quad$ (3)

Results

Model	Valid mIoU	Train mIoU
norm-conv	22.18%	31.03%
def-conv-v1	22.98% (+0.80%)	32.09% (+1.06%)
def-conv-v2	23.59% (+1.41%)	32.92% (+1.89%)

Conclusion

- Deformable Convolutions works!
  - Gives +1.4% mIoU on validation
  - Adds 300-400k params, which is ~1% of the total number of parameters
  - Learning is a little slower
- My offset visualizations are not impressive ;(
- I underestimated the task:
  - ADE20k is hard dataset with 150 classes
  - ResNet50 is huge network with 33M params, which took 3-4h to train
- Always select hyperparameters from papers
- There were two major bugs in the code that forced me to retrain the models
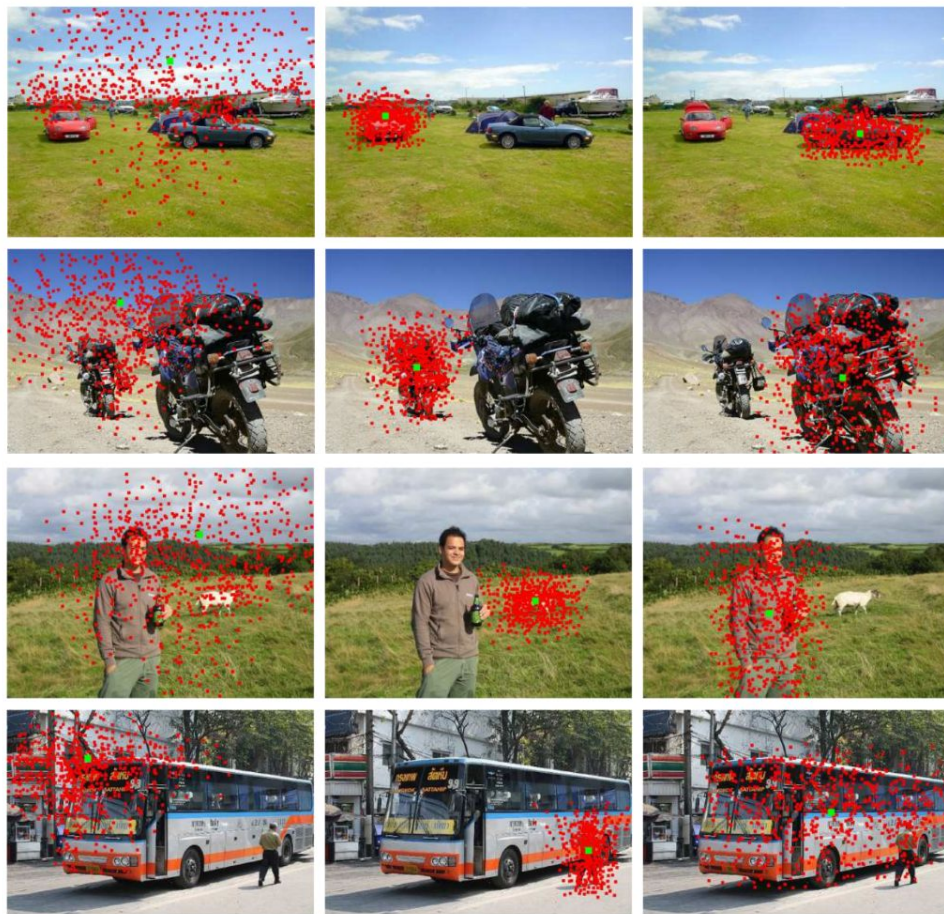
Visualization of the offsets from the original paper