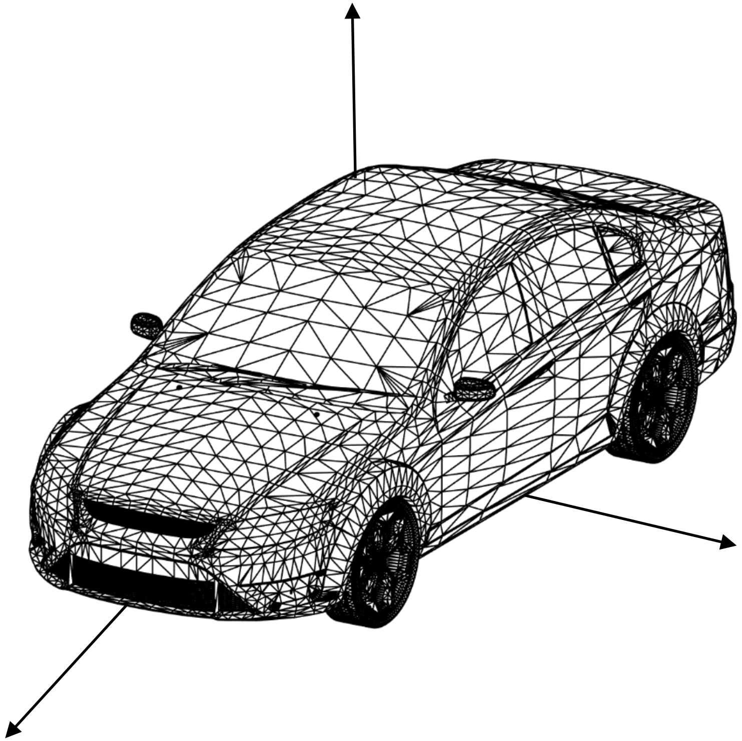
3D Pose Regression using Convolutional Neural Networks

Siddharth Mahendran, Haider Ali, and René Vidal
Center for Imaging Science
Johns Hopkins University

Problem Statement

6D Task: given a single 2D image, estimate 6D object pose



Problem Statement

6D Task: given a single 2D image, estimate 6D object pose

2D detection has experienced significant progress over the past few years

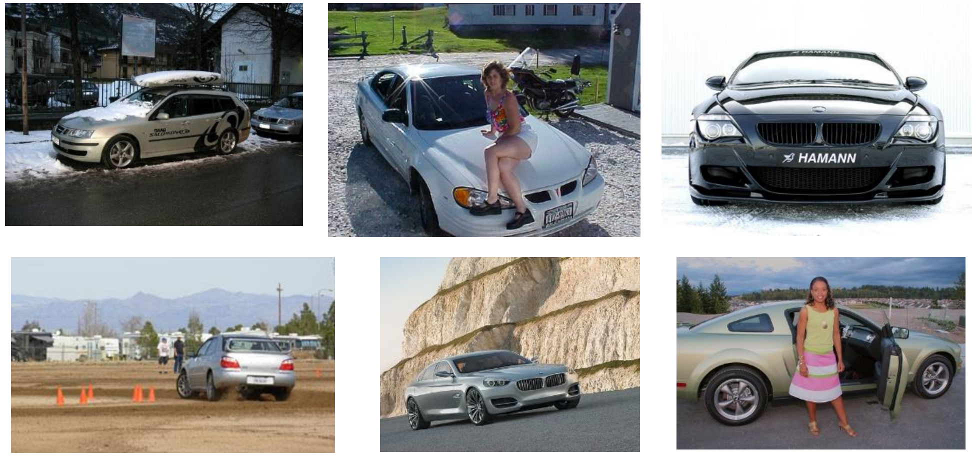
Assume a 2D bounding box returned by an oracle or an object detector

3D Task: Given a 2D image and a 2D bounding box around an object in the image, predict the 3D orientation of the object

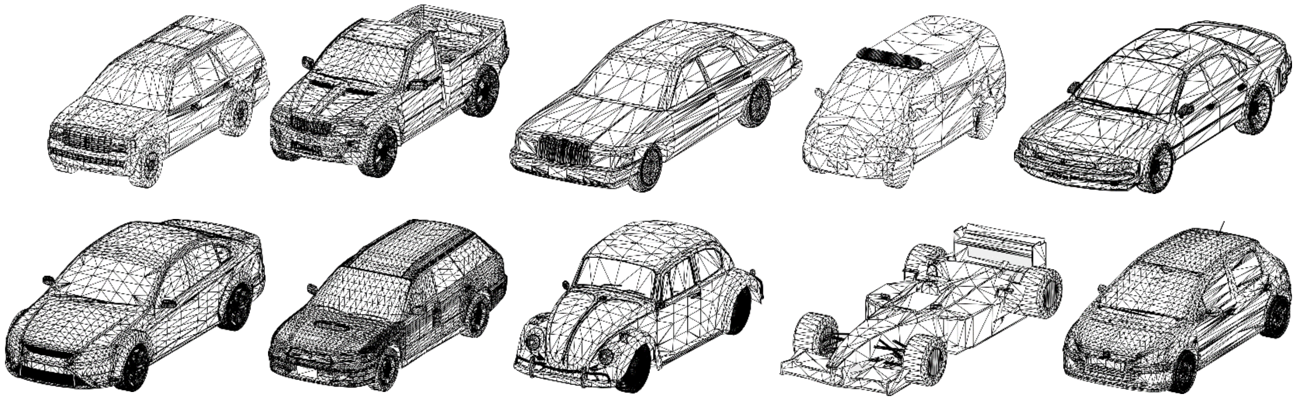
Problem Formulation

$\rightarrow$ $\rightarrow$ R

Ill Posed !!



Learn from training examples



Pose annotations with aligned models

Problem Formulation

CNN → R

What is the network architecture ?

What representation and loss function to use ?

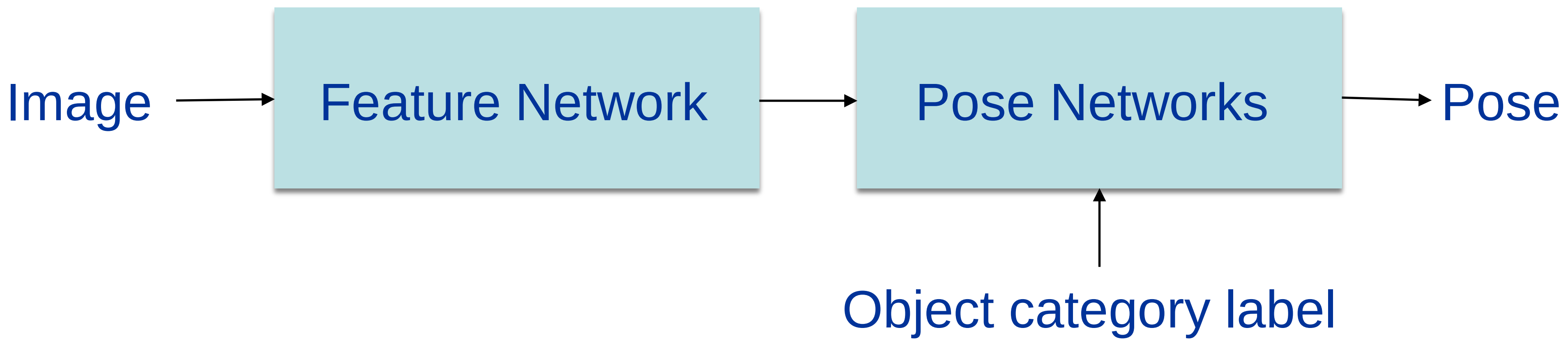
What data to use ?
Any data augmentation ?

Paper Contributions

	Prior work	This work
Problem formulation	Pose classification	Pose regression
Representation	Discretized angle bins	Axis-angle / Quaternion
Loss function	Cross-entropy loss	Geodesic loss
Data augmentation	2D jittering [1] Rendered images [2]	3D pose jittering + Rendered images

[1] S. Tulsiani and J. Malik, *Viewpoints and Keypoints*, CVPR 2015
[2] H. Su, C. Qi, Y. Li, and L. Guibas, *Render for cnn: Viewpoint estimation in images using cnns trained with rendered 3d model views*, ICCV 2015

Network Architecture for 3D Pose Task



Feature Network: VGG-M [1] upto FC6

Pose Network: (per object category) 3 Fully Connected layers with Batch Normalization and ReLU activations

[1] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman. Return of the devil in the details: Delving deep into convolutional nets. BMVC 2014

Representations and Loss Functions for 3D Pose Task

Exploit underlying structure of rotation matrices !

$$R \in SO(3) \doteq \{X \in \mathbb{R}^{3\times 3} | X^T X = I_3, \det(X) = 1\}$$

Rotation by an angle about an axis

$$R = \text{expm}(\theta[v]_\times)$$

Axis-angle

$$R \leftrightarrow y = \theta v$$

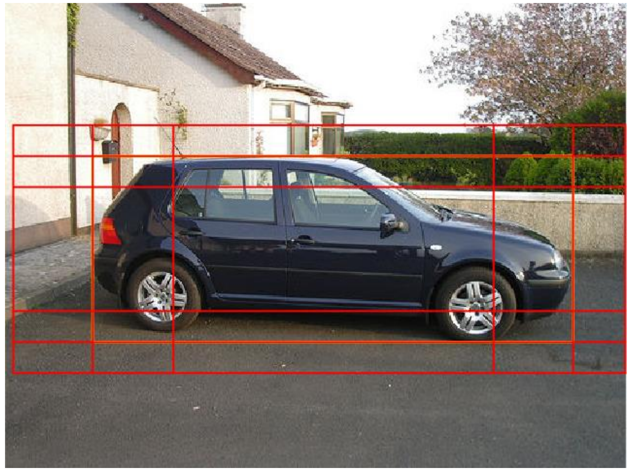
$$\mathcal{L}(y_1, y_2) = \cos^{-1}\left[\frac{\text{trace}(R_1^T R_2) - 1}{2}\right]$$

Quaternion

$$R \leftrightarrow q = \left(\cos\frac{\theta}{2}, \sin\frac{\theta}{2} v\right)$$

$$\mathcal{L}(q_1, q_2) = 2\cos^{-1} |\langle q_1, q_2 \rangle|$$

Data Augmentation for 3D Pose Task



2D Pose jittering

Unknown perturbations in 3D pose !!

Perturbation around Z-axis: $R(az, el, ct \pm \delta ct)$

Perturbation around X-axis: $R(az \pm \delta az, el, ct)$

3D Pose jittering

Experimental Setup

- Dataset: Pascal3D+ (release 1.1)
  - ImageNet and Pascal VOC2012 images for 12 object categories
    - Training set: Imagenet-trainval images,
    - Validation set: Pascal-train images
    - Testing set: Pascal-val images

- Data augmentation:
  - 3D pose jittering – 162 samples per image
    - Perturbations around X-axis (x9) : -2:0.5:2
    - Perturbations around Z-axis (x9) : -4:1:4
    - Flips (x2)
  - Rendered images [1]

- Training:
  - Adam optimizer with learning rate schedule
  - Implemented in Keras with TensorFlow backend

Evaluation metric:

$$\mathcal{L}(R_1, R_2) = \frac{\| \log(R_1 R_2^T) \|}{\sqrt{2}}$$

[1] H. Su, C. Qi, Y. Li, and L. Guibas, *Render for cnn: Viewpoint estimation in images using cnns trained with rendered 3d model views*, ICCV 2015

Results

Median angle error between predicted and ground-truth rotation matrices

	aero	bike	boat	bottle	bus	car	chair	dtable	mbike	sofa	train	tv	mean
Ours: axis-angle	13.97	21.07	35.52	**8.99**	4.08	7.56	21.18	17.74	17.87	12.70	8.22	15.68	15.38
Ours: quaternion	14.53	22.55	35.78	9.29	4.28	8.06	19.11	30.62	18.80	13.22	7.32	16.01	16.63

Performance on ground-truth bounding boxes for un-occluded and un-truncated objects

Ours: axis-angle detected	14.71	21.31	45.07	9.47	4.20	8.93	26.36	20.70	19.16	18.80	8.72	15.65	17.76

Performance on bounding boxes returned by Faster R-CNN [3]

[1] S. Tulsiani and J. Malik, *Viewpoints and Keypoints*, CVPR 2015
[2] H. Su, C. Qi, Y. Li, and L. Guibas, *Render for cnn: Viewpoint estimation in images using cnns trained with rendered 3d model views*, ICCV 2015
[3] S. Ren, K. He, R. Girshick, and J. Sun. Faster RCNN: Towards real-time object detection with region proposal networks. Arxiv 2015

Conclusion

We designed a Convolutional Neural Network framework for the task of 3D Pose regression with :

- Suitable representation of the space of 3D rotation matrices: axis-angle and quaternion

- Appropriate geodesic loss on the space of rotation matrices

- Relevant data augmentation strategy, 3D pose jittering based on applying homographies to the images

Acknowledgements

- Collaborators

Siddharth Mahendran

Haider Ali

- Funding
  - NSF 1527340

Vision Lab @ Johns Hopkins University

http://www.vision.jhu.edu

Center for Imaging Science @ Johns Hopkins University

http://www.cis.jhu.edu

Thank You!