Probabilistic Regression using Convolutional Neural Networks*

Muhammad Asad
Senior Research Engineer
Imagination Technologies

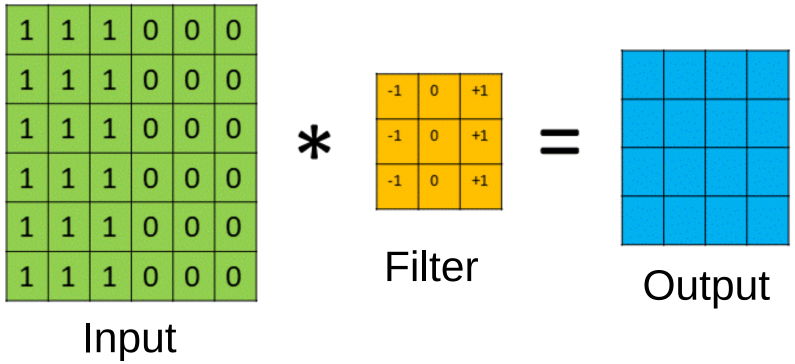
masadcv@gmail.com

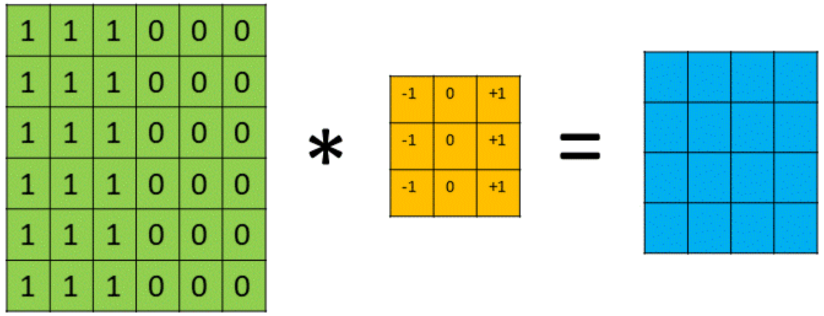
linkedin.com/in/masadcv/

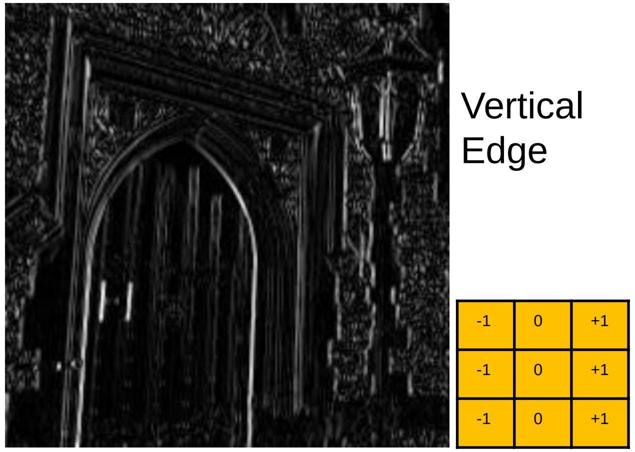
*Based on collaborative work at City, University of London

Based on:
M. Asad, R. Basaru, SMM. Al-Arif, G. Slabaugh. "**PROPEL: Probabilistic Parametric Regression Loss for Convolutional Neural Networks.**" Accepted at *International Conference on Pattern Recognition (ICPR). 2020*. Preprint at: https://arxiv.org/abs/1807.10937

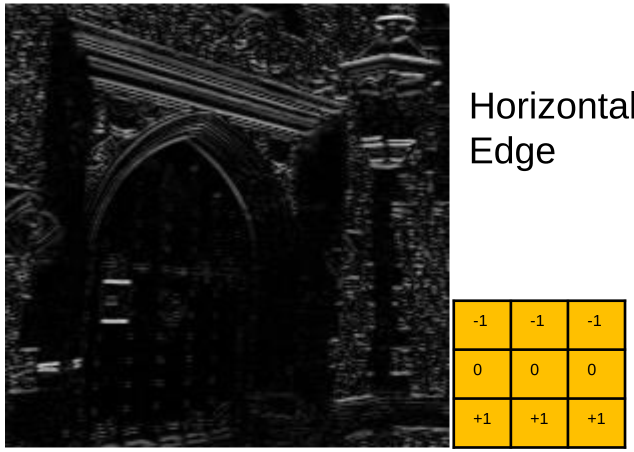
What are Convolutions?







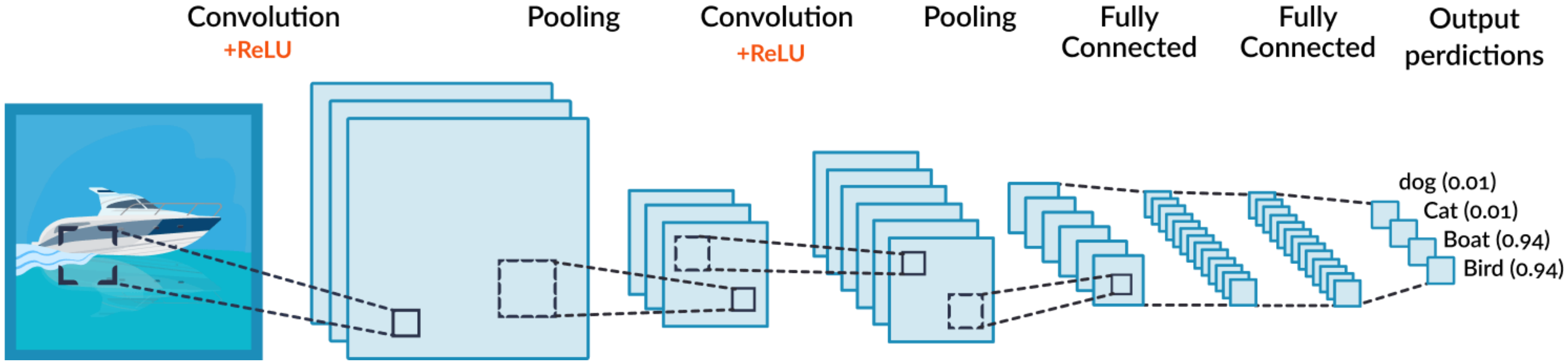
Vertical Edge



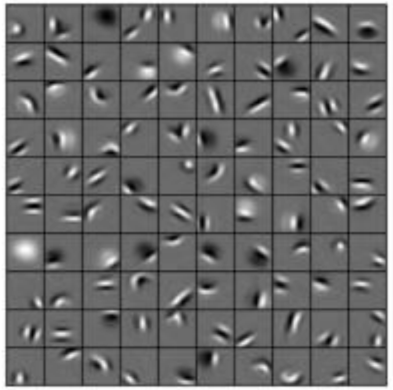
Horizontal Edge

Source: https://www.youtube.com/watch?v=XuD4C8vJzEQ

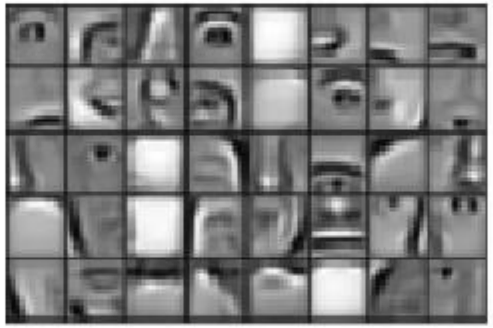
What are Convolutional Neural Networks?



Low level



Mid level



High level

Source: https://missinglink.ai/guides/convolutional-neural-networks/convolutional-neural-network-tutorial-basic-advanced/
https://developer.nvidia.com/blog/deep-learning-nutshell-core-concepts/#feature-engineering

Applications of Convolutional Neural Networks

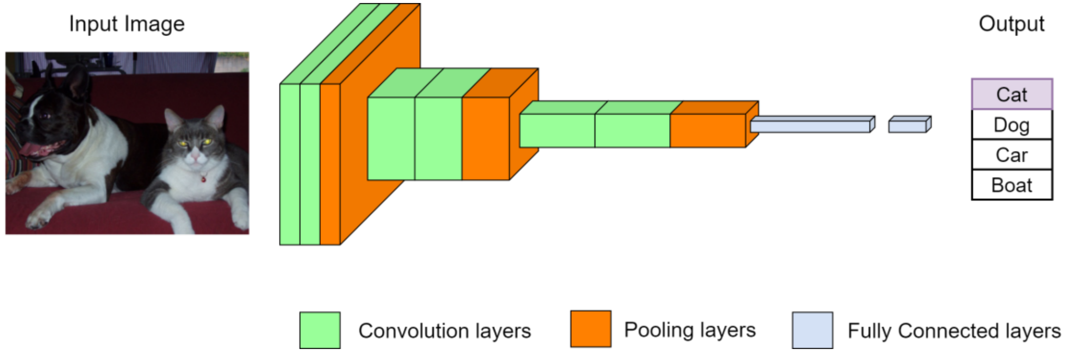


Image → Object Class

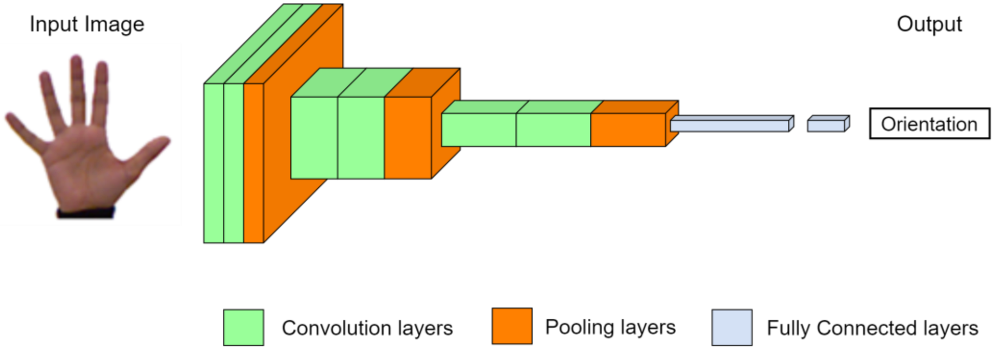


Image → Real Number

This talk!

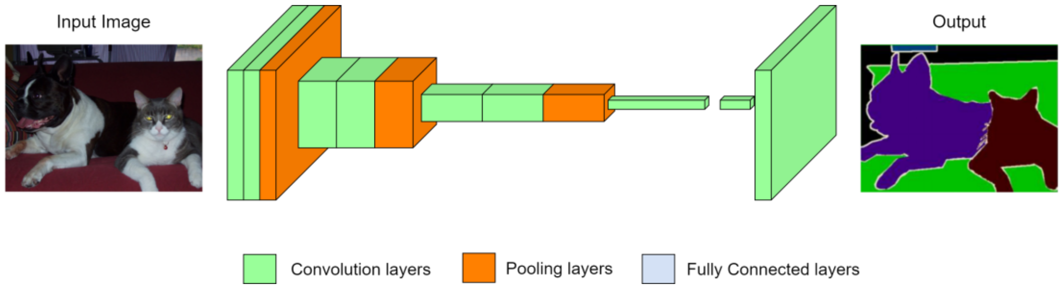
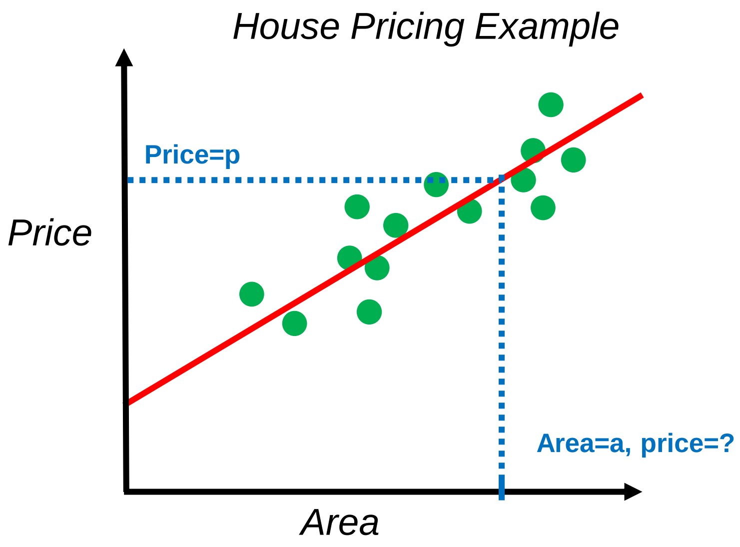


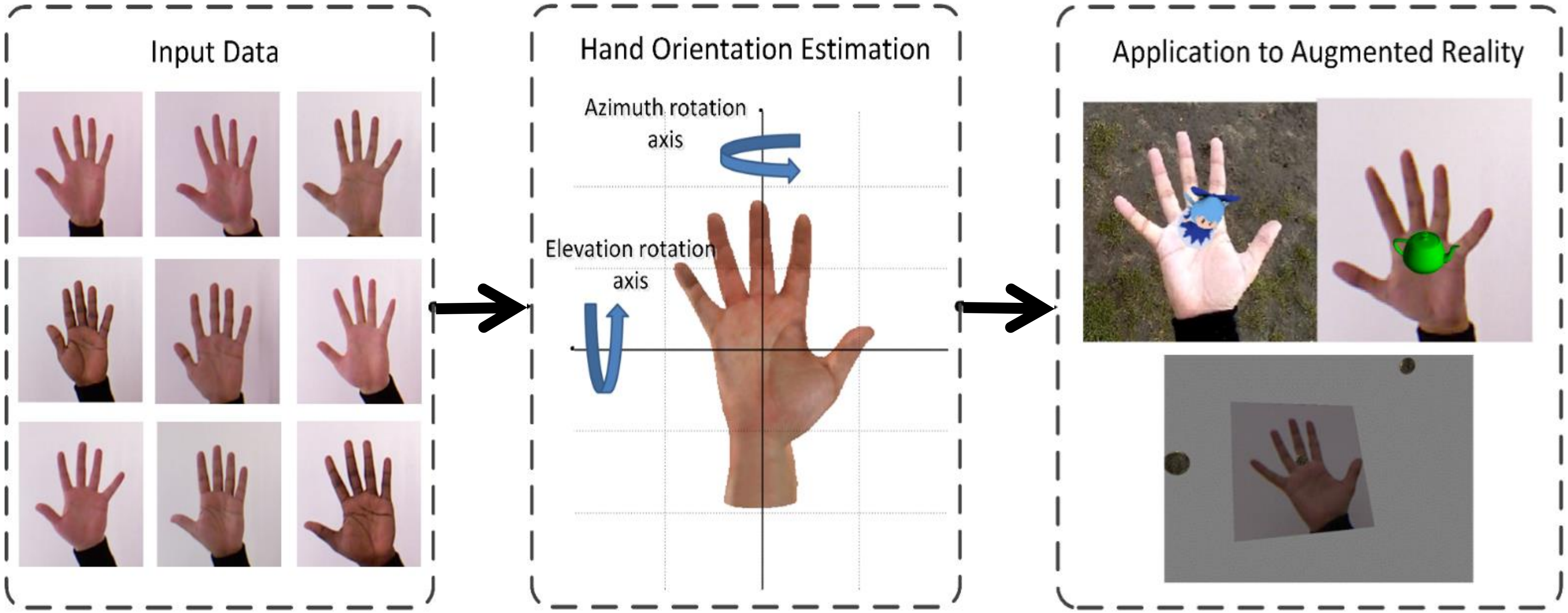
Image → Image

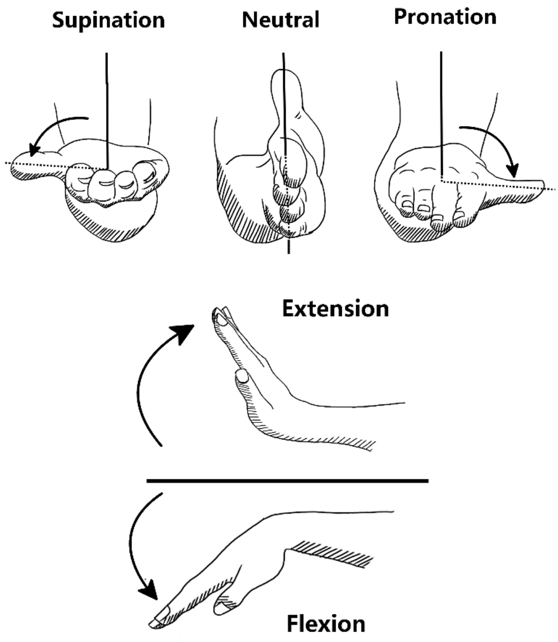


Source: https://missinglink.ai/guides/convolutional-neural-networks/convolutional-neural-network-tutorial-basic-advanced/

Problem Definition

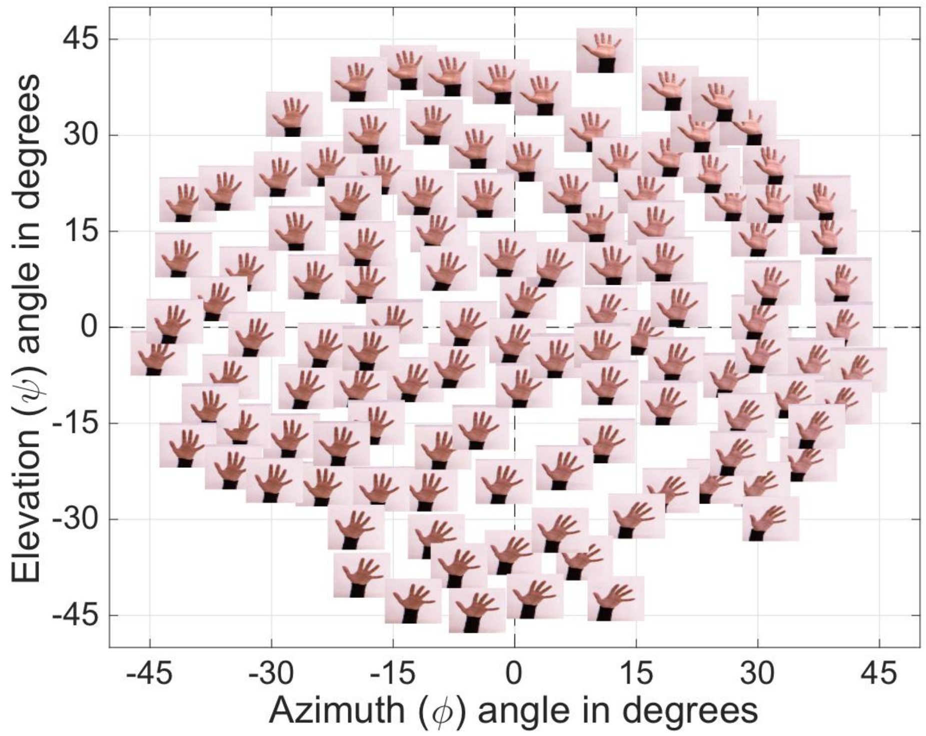
Can we use a machine learning model to learn the mapping of 2D images onto 3D hand orientation?





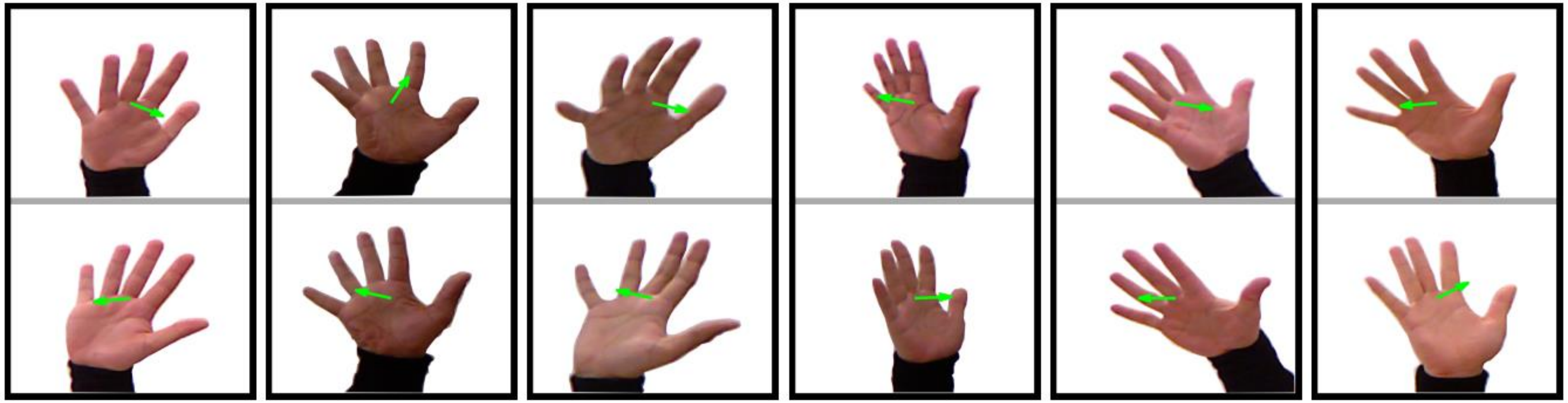
Hand Orientation Dataset

- Collected 9414 planar 2D hand images with annotated 3D orientation angles
- 22 participants with different hand shape, size and style

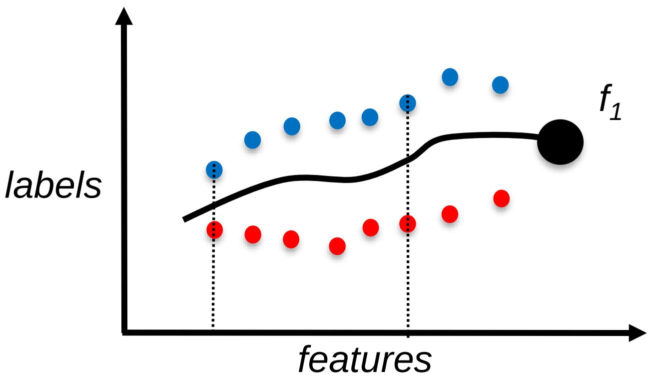


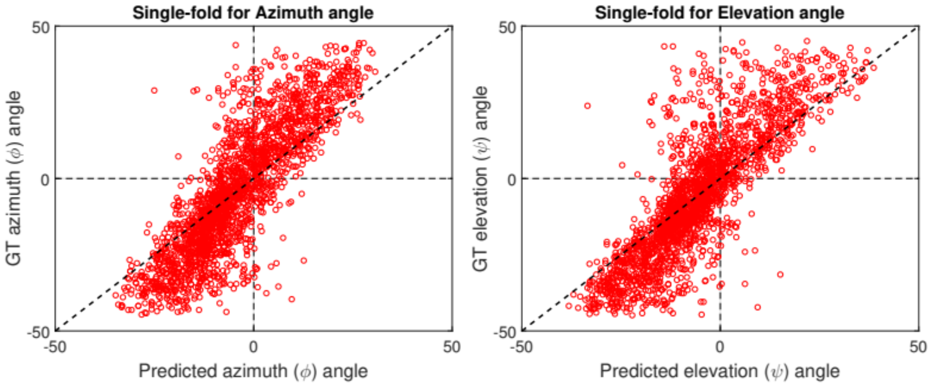
Ambiguity within the Dataset

- Symmetry problem: opposite orientation ←→ similar hand shapes



- Existing regression methods *try to fit* into the data (high bias) [1, 2]





- Motivates the need for probabilistic regression that can enable handling multiple-hypotheses

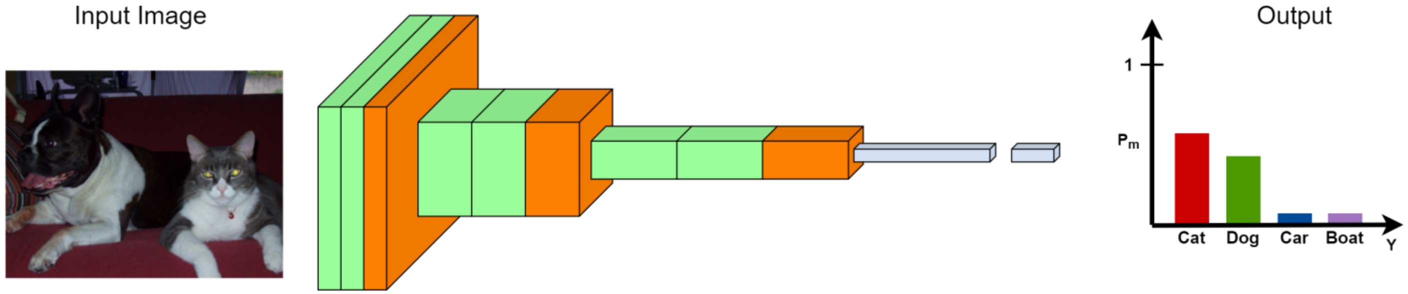
[1] M. Asad, G. Slabaugh. "**Learning marginalization through regression for hand orientation inference.**" *CVPR Workshop*. 2016.
[2] M. Asad, G. Slabaugh. "**SPORE: Staged Probabilistic Regression for Hand Orientation Inference.**" Computer Vision and Image Understanding (CVIU). 2017.

Existing Probabilistic Learning with CNNs

Let P_{gt} be ground truth target distribution, CNN learns P_m using loss functions:

Task	Network	Loss

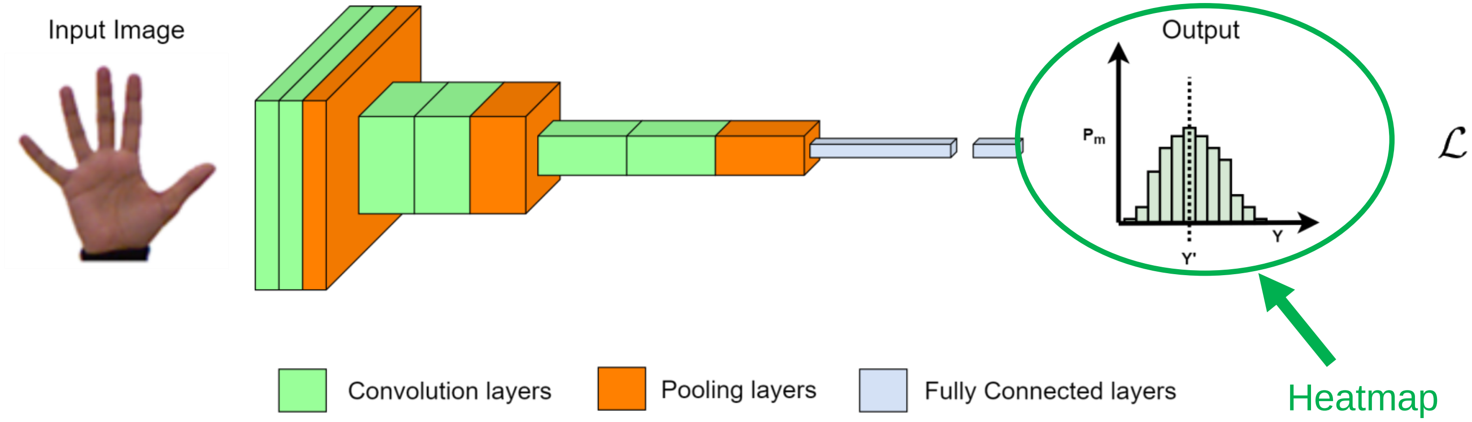
Classification



$$\mathcal{L}(P_{gt}, P_m) = -\sum_{i \in I} P_{gt}^i \log(P_m^i)$$

Cross Entropy

Regression



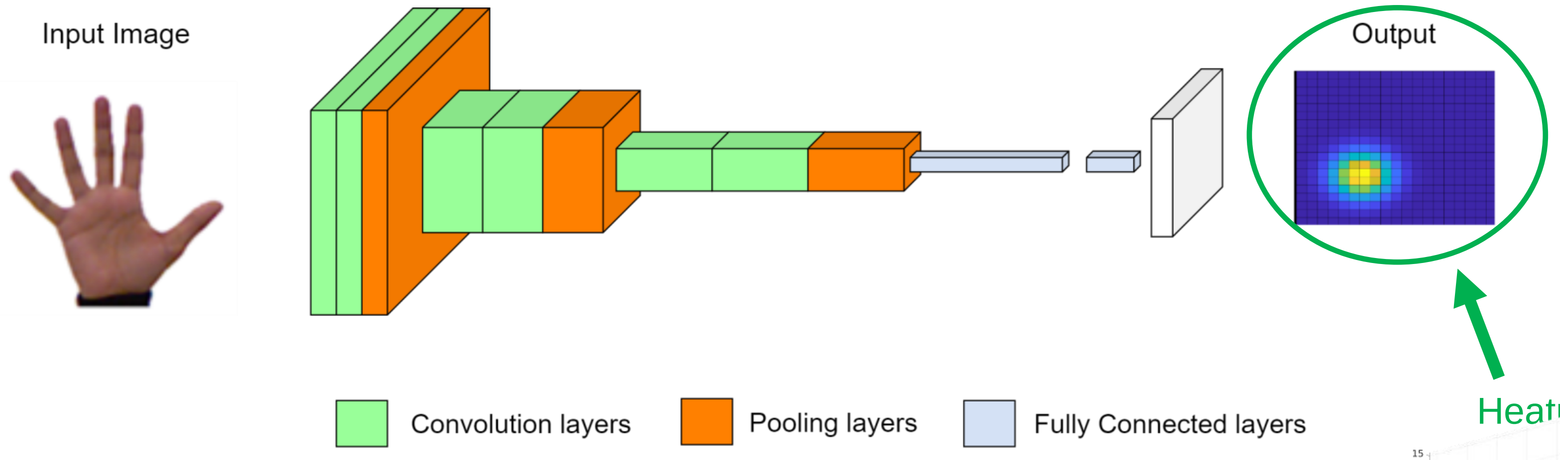
$$\mathcal{L}(P_{gt}, P_m) = \frac{1}{2} \sum_{i \in I} (P_{gt}^i - P_m^i)^2$$

Mean Squared Error (MSE)

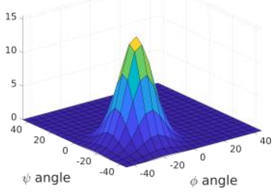
Existing Probabilistic Regression using CNNs

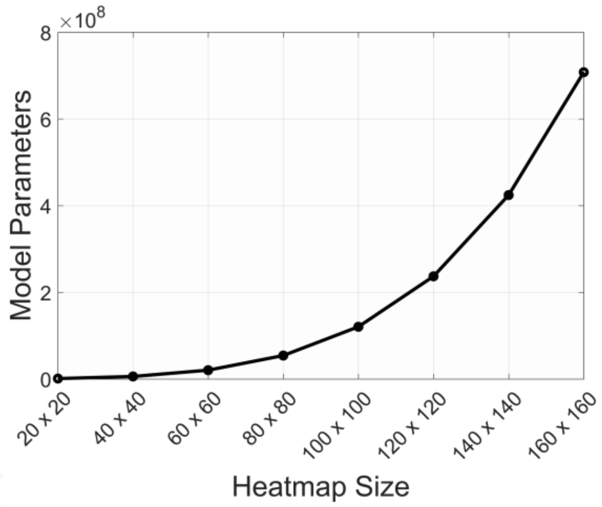
- Heatmap → higher dimensional grid with probabilities for each point
  → requires larger model (as compared to directly learning target)

better accuracy → increase heatmap size → increase model complexity → overfitting



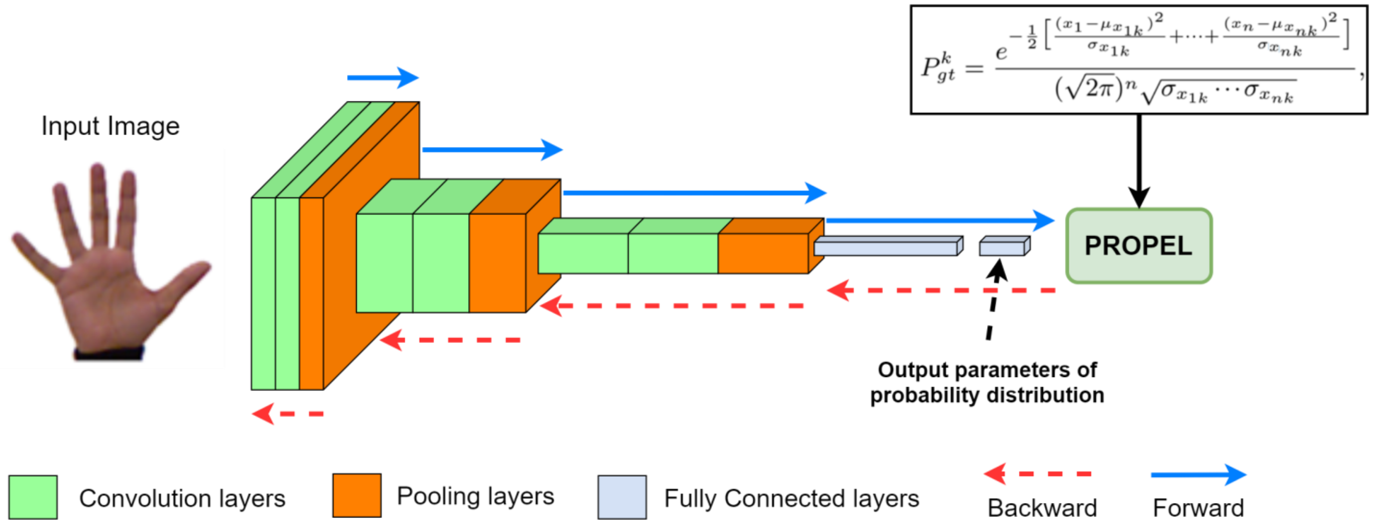
Heatmap





PRObabilistic Parametric rEgression Loss (PROPEL) [3]

- Can we learn probabilities as parameters of a probability distribution function?
- Propose novel loss function → learns parameters of a mixture of Gaussian distribution
  - Fully-differentiable, analytic closed form solution, works with standard optimizers e.g. RMSProp, ADAM
  - Enables learning multi-modal mixture of Gaussian distribution
  - Generalizes better with less model parameters → less overfitting → faster optimization



Learn these parameters directly

$$P_m = \frac{1}{I}\sum_{i=1}^{I} P_i = \frac{1}{I}\sum_{i=1}^{I} \frac{e^{-\frac{1}{2}\left[\frac{(x_1-\mu_{x_{1i}})^2}{\sigma_{x_{1i}}}+\cdots+\frac{(x_n-\mu_{x_{ni}})^2}{\sigma_{x_{ni}}}\right]}}{(\sqrt{2\pi})^n\sqrt{\sigma_{x_{1i}}\cdots\sigma_{x_{ni}}}},$$

[3] M. Asad, R. Basaru, SMM. Al-Arif, G. Slabaugh. "**PROPEL: Probabilistic Parametric Regression Loss for Convolutional Neural Networks.**" Accepted at *International Conference on Pattern Recognition (ICPR). 2020*.

PROPEL Definition [3]

- Let $\mathbf{x} = \{x_1, x_2, \cdots, x_n\}^\intercal \in \mathbb{R}^n$ define target prediction space
- PROPEL is defined as (using metric from [*]):

$$L = -\log\left[\frac{2\int P_{gt}P_m\, d\underline{\mathbf{x}}}{\int ({P_{gt}}^2 + {P_m}^2)\, d\underline{\mathbf{x}}}\right],$$

$$P_{gt}^k = \frac{e^{-\frac{1}{2}\left[\frac{(x_1-\mu_{x_{1k}})^2}{\sigma_{x_{1k}}} + \cdots + \frac{(x_n-\mu_{x_{nk}})^2}{\sigma_{x_{nk}}}\right]}}{(\sqrt{2\pi})^n\sqrt{\sigma_{x_{1k}}\cdots\sigma_{x_{nk}}}},$$

$$P_m = \frac{1}{I}\sum_{i=1}^{I} P_i = \frac{1}{I}\sum_{i=1}^{I}\frac{e^{-\frac{1}{2}\left[\frac{(x_1-\mu_{x_{1i}})^2}{\sigma_{x_{1i}}} + \cdots + \frac{(x_n-\mu_{x_{ni}})^2}{\sigma_{x_{ni}}}\right]}}{(\sqrt{2\pi})^n\sqrt{\sigma_{x_{1i}}\cdots\sigma_{x_{ni}}}},$$

P_{gt} : n-dimensional ground truth PDF

P_m : mixture of Gaussian learned model PDF

- Partial derivatives for optimizing each parameter in model PDF P_m:

$$\frac{\partial L}{\partial \mu_{x_{ni}}} = -\frac{1}{T1}\left[\frac{\partial G(P_{gt}, P_i)}{\partial \mu_{x_{ni}}}\right] + \frac{1}{T2}\left[\frac{2}{I^2}\sum_{i<j}^{I}\frac{\partial G(P_i, P_j)}{\partial \mu_{x_{ni}}}\right],$$

$$\frac{\partial L}{\partial \sigma_{x_{ni}}} = -\frac{1}{T1}\left[\frac{\partial G(P_{gt}, P_i)}{\partial \sigma_{x_{ni}}}\right] + \frac{1}{T2}\left[\frac{1}{I^2}\frac{\partial H(P_i)}{\partial \sigma_{x_{ni}}} + \frac{2}{I^2}\sum_{i<j}^{I}\frac{\partial G(P_i, P_j)}{\partial \sigma_{x_{ni}}}\right]$$

[*] S. Giorgos, et al. "**An analytic distance metric for Gaussian mixture models with application in image retrieval.**" *International Conference on Artificial Neural Networks (ICANN)*. 2005.
[3] M. Asad, R. Basaru, SMM. Al-Arif, G. Slabaugh. "**PROPEL: Probabilistic Parametric Regression Loss for Convolutional Neural Networks.**" Accepted at *International Conference on Pattern Recognition (ICPR). 2020*.

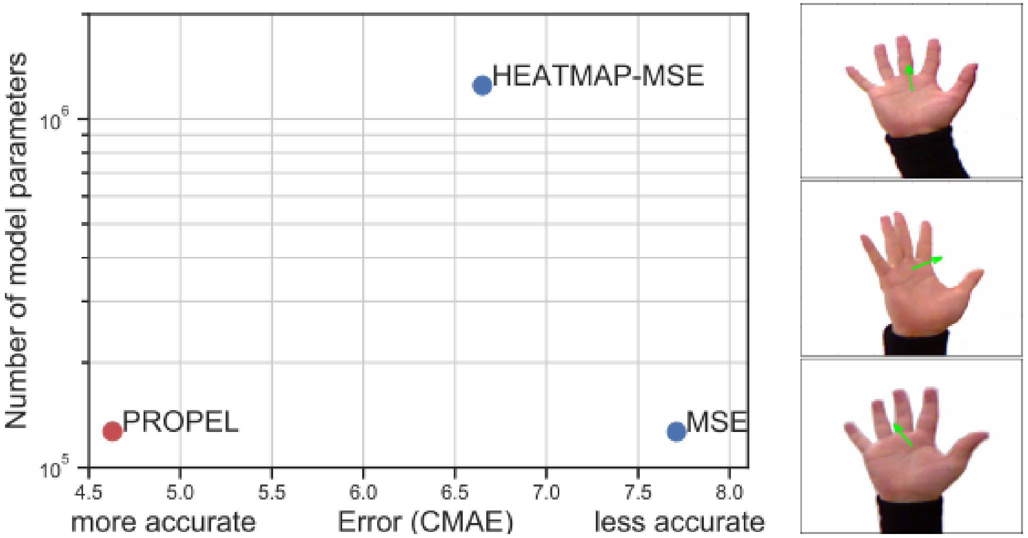
Why not use KL-Divergence/Bhattacharyya Distance?

$$D_{KL}(P_{gt}||P_m) = \int P_{gt} \log\left(\frac{P_{gt}}{P_m}\right)\, d\underline{\mathbf{x}},$$
$$= \int P_{gt}\log(P_{gt})\, d\underline{\mathbf{x}} - \int P_{gt}\log(P_m)\, d\underline{\mathbf{x}},$$
$$= \int P_{gt}\log(P_{gt})\, d\underline{\mathbf{x}} - \int P_{gt}\log\left(\frac{1}{I}\sum_{i=1}^{I} P_i\right)\, d\underline{\mathbf{x}}.$$

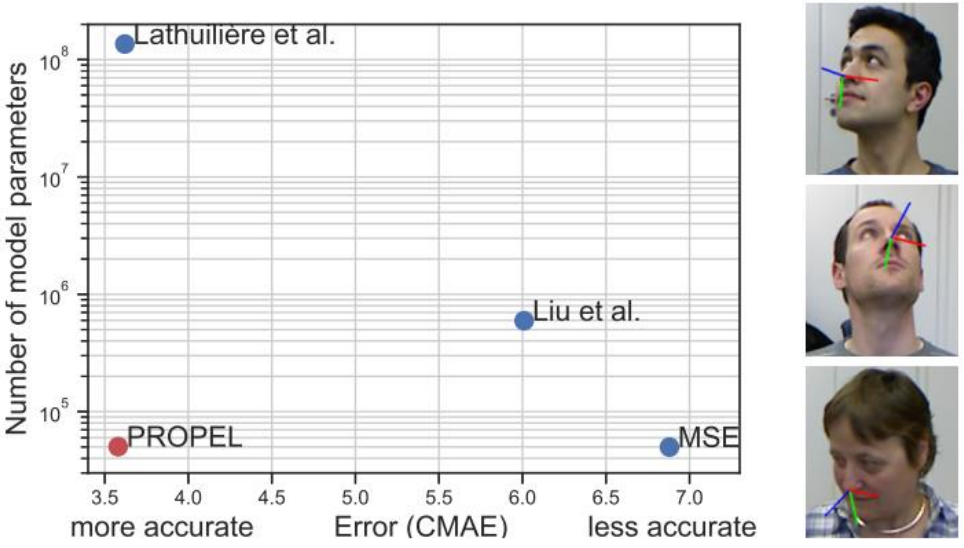
$$D_{BC}(P_{gt}||P_m) = -\log\int\sqrt{P_{gt}P_m}\, d\underline{\mathbf{x}},$$
$$= -\log\int\sqrt{P_{gt}\left(\frac{1}{I}\sum_{i=1}^{I} P_i\right)}\, d\underline{\mathbf{x}}.$$

No analytic solution to integrals!!

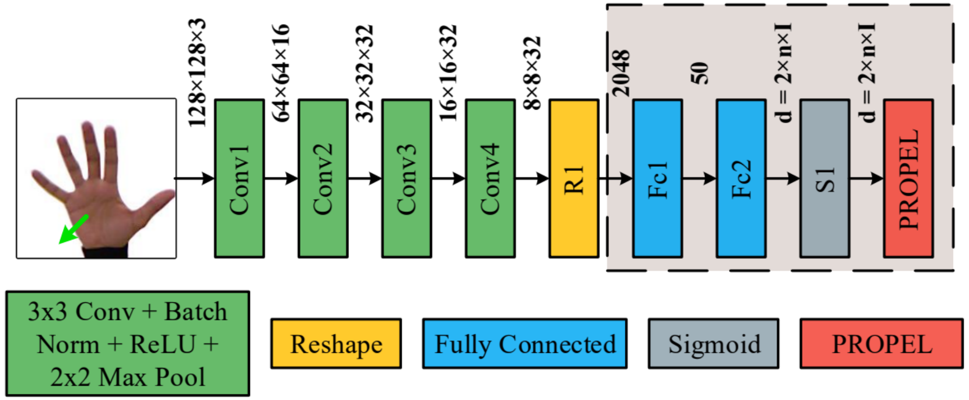
Experimental Validation: Accuracy + Efficiency

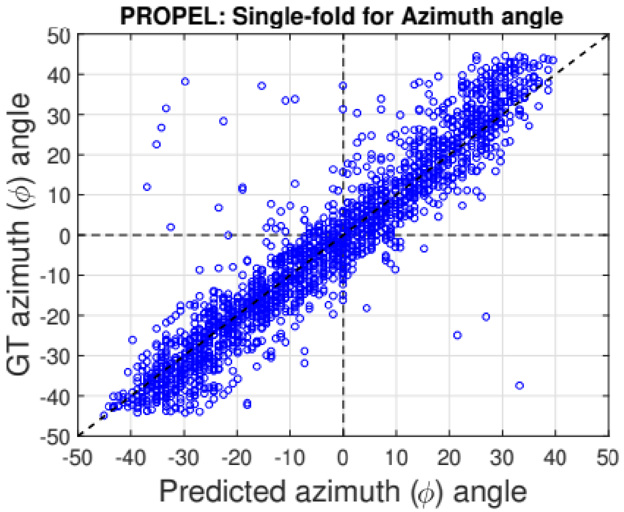


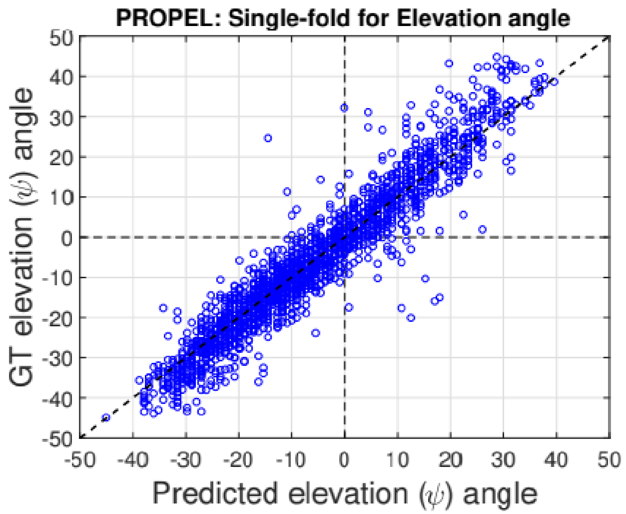
(a) Hand orientation estimation



(b) Head orientation estimation







Thank you for listening

- Importance of Probabilistic Regression
- Limitations with existing heatmap based CNN Regression
- Learning parameters of probability distribution