JOHANNES GUTENBERG
UNIVERSITÄT MAINZ

Deep Feature Interpolation for Image Content Change

Steven Lang, Manfred Faldum

February 8, 2017

Motivation

Basic Idea

Motivation

What we want

Motivation

What we want

- Add attribute in pixel space

Motivation

What we want

- Add attribute in pixel space
- New idea: add attribute in deep feature space

Basic Idea

How to get the attribute in deep feature space?

Basic Idea

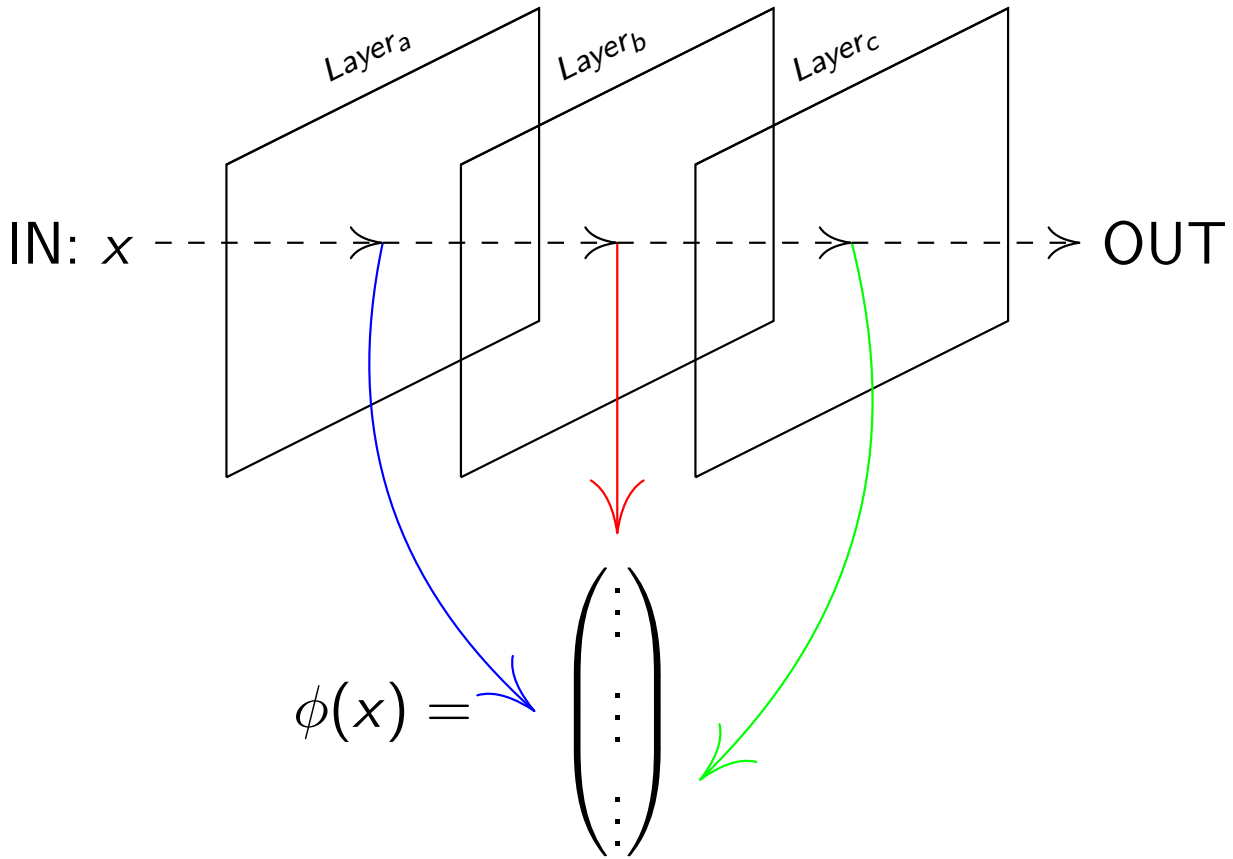
How to get the attribute in deep feature space?

- Let $\phi(x)$ be the mapping from pixel space into deep feature space by concatenating an arbitrary number of layers

Basic Idea

How to get the attribute in deep feature space?

- Let $\phi(x)$ be the mapping from pixel space into deep feature space by concatenating an arbitrary number of layers



Basic Idea

- Take k nearest neighbor images with existing attribute: S^+
- Take k nearest neighbor images with missing attribute: S^-

Basic Idea

- Take k nearest neighbor images with existing attribute: S^+
- Take k nearest neighbor images with missing attribute: S^-

- $\phi^+ = \phi(S^+)$ and $\phi^- = \phi(S^-)$
- Build the mean $\overline{\phi^+}$ and $\overline{\phi^-}$

Basic Idea

- Take k nearest neighbor images with existing attribute: S^+
- Take k nearest neighbor images with missing attribute: S^-

- $\phi^+ = \phi(S^+)$ and $\phi^- = \phi(S^-)$
- Build the mean $\overline{\phi^+}$ and $\overline{\phi^-}$

- Representation of attribute: $w = \overline{\phi^+} - \overline{\phi^-}$

Basic Idea

How to get the output picture?

Basic Idea

How to get the output picture?

- $\phi(z) = \phi(x) + \alpha w$
- Reverse mapping of $\phi(z)$ into pixel space:

Basic Idea

How to get the output picture?

- $\phi(z) = \phi(x) + \alpha w$
- Reverse mapping of $\phi(z)$ into pixel space:
  - $\tilde{z} = \underset{\tilde{z}}{\operatorname{argmin}} \frac{1}{2}||\phi(z) - \phi(\tilde{z})||_2^2 + \lambda R_\beta(\tilde{z})$
  - with $R_\beta(\tilde{z}) = \sum_{i,j}((\tilde{z}_{i,j+1} - \tilde{z}_{i,j})^2 + (\tilde{z}_{i+1,j} - \tilde{z}_{i,j})^2)^{\frac{\beta}{2}}$

Technical

- Model: VGG19 pretrained on IMAGENET dataset
- $\phi(x)$: third, fourth and fifth Relu Layer
- Regularization parameters: $\beta = 2$, $\lambda = 0.001$
- Test set: labeled faces in the wild (LFW dataset)
- Optimizer: Adam
- KNN: on discrete attribute features (mustache, smiling, ...)
- Normalization: $\phi(x) = \frac{\phi(x)}{||\phi(x)||}$, $w = \frac{w}{||w||}$

Results: Eyeglasses

$\alpha = 0.4,\ k = 100$

Results: Sunglasses

$\alpha = 0.4,\ k = 100$

Results: Mouth open

$\alpha = 0.4,\ k = 100$

Results: Female

$\alpha = 0.4,\ k = 100$

Results: Heavy makeup

$\alpha = 0.4,\ k = 100$

Results: Smiling

$\alpha = 0.4,\ k = 100$

Results: Mustache

$\alpha = 0.4,\ k = 100$

Results: Asian

$\alpha = 0.4,\ k = 100$

Varying the parameters: Senior

$k = 100$, $\alpha \in [0.3, 0.4, \ldots, 1.0]$

Varying the parameters: Senior

$k \in [1, 10, 20, 50, 75, 100, 150, 200]$, $\alpha = 0.5$

Varying the parameters: Smiling

$k = 100$, $\alpha \in [0.3, 0.4, \ldots, 1.0]$

Varying the parameters: Smiling

$k \in [1, 10, 20, 50, 75, 100, 150, 200]$, $\alpha = 0.5$

Outlook

- Multiple features at once: $\phi(z) = \phi(x) + \alpha \sum_i w_i$
- Nearest Neighbor in deep feature space
- Choose more/less/other layers for ϕ
- Change normalization: $w = w \frac{||\phi(x)||}{||w||}$