Object and Action Detection from a Single Example

Peyman Milanfar*
EE Department
University of California, Santa Cruz

*Joint work with Hae Jong Seo

AFOSR Program Review, June 4-5, 2009
Take a look at this:
See it here?
How about here?
Or here?
Single Example, No Training!

(Most) people can find the Dragon Fruit from one look.

Even if they’ve never seen it before.
Outline

I. Motivation
II. Overview
III. Object Detection
IV. Action Detection
V. Conclusion and Future work
Fundamental Problems in Machine Vision

Develop a unified framework that can robustly detect objects/actions of interest within images/videos without training.

1) Whether objects (actions) are present or not,
2) How many objects (actions)?
3) Where are they located?
Challenges in Detection

- **Objects**
  - Background clutter
  - Scale
  - Pose
  - Intra-class variation

- **Actions**
  - UNDERWATER
  - RAINDROP
  - MEDICAL imaging
  - Noise
  - Blur

Besides, Contexts:

Degradation:

1) different clothes,
2) different illumination,
3) different background
4) action speed
Outline

I. Motivation

II. System Overview

III. Object Detection

IV. Action Detection

V. Conclusion and Future work
Object Detection using Local Regression Kernels

- Local Steering Kernels as *Descriptors*
- Using a *single* example

“Resemblance Map”

Detected Similar Objects

Query
Object Detection System Overview

- **Stage 1**: Compute local steering kernels Descriptors
  - $Q \rightarrow W_Q$
  - $T \rightarrow W_T$

- **Stage 2**: PCA
  - $A_Q' W_Q$
  - $A_Q' W_T$
  - Compute feature images

- **Stage 3**:
  - 2) Significance Tests
  - 3) Non-maxima Suppression
  - Final result
  - 1) Resemblance Map (RM) using Matrix Cosine Similarity
    - $RM : f(\rho)_i$

H. Seo and P. Milanfar, “Training-free, Generic Object Detection using Locally Adaptive Regression Kernels”, Accepted for publication in IEEE Transactions on Pattern Analysis and Machine Intelligence
Stage 1: Calculation of Local Descriptors

\[ K(x_l - x) = \sqrt{\frac{\det(C_l)}{2h^2}} \exp \left\{ -\frac{(x_l - x)'C_l(x_l - x)}{2h^2} \right\} \]

\[ W(x_l - x) = \frac{K(x_l - x)}{\sum_{l=1}^{P} K(x_l - x)} \]

SVD
Robustness of LSK Descriptors

\[ W_Q(x_l - x) \]

1. Original image
2. Brightness change
3. Contrast change
4. WGN sigma = 10

Milanfar et al. EE Dept, UCSC
System Overview: Stage 2

Query

Target

stage 1

\[ W_Q, W_T \]

\[ A_Q, A'_Q \]

\[ F_Q, F_T \]

\[ A'_Q W_Q, A'_Q W_T \]

\[ \text{PCA} \]

\[ \text{compute feature images} \]

\[ \text{local steering kernels} \]

stage 2

stage 3

2) Significance Tests
3) Non-maxima Suppression

Final result

1) Resemblance Map (RM) using Matrix Cosine Similarity

\[ RM : f(\rho)_i \]
Stage 2: Feature Extraction from Descriptors

Apply **PCA** to $W_Q$ for dimensionality reduction

$\rightarrow$ Retain the $d$ largest principal components $A_Q \in \mathbb{R}^{P \times d}$

$\rightarrow$ Project $W_Q$ and $W_T$ onto $A_Q$

$$ F_Q = [f_Q^1, \ldots, f_Q^n] = A_Q^\prime W_Q $$

$$ F_T = [f_T^1, \ldots, f_T^n] = A_Q^\prime W_T $$

Milanfar et al. EE Dept, UCSC
Stage 2: Salient features after PCA

Object: Helicopter

<table>
<thead>
<tr>
<th>Eigenvectors</th>
<th>Query features</th>
<th>Target features</th>
</tr>
</thead>
<tbody>
<tr>
<td>LSK $W_Q(x_l - x; 2)$</td>
<td>Query $Q$</td>
<td>Target $T$</td>
</tr>
<tr>
<td>1st eigenvector $A_Q(1)$</td>
<td>$F_Q(1)$</td>
<td>$F_T(1)$</td>
</tr>
<tr>
<td>2nd eigenvector $A_Q(2)$</td>
<td>$F_Q(2)$</td>
<td>$F_T(2)$</td>
</tr>
<tr>
<td>3rd eigenvector $A_Q(3)$</td>
<td>$F_Q(3)$</td>
<td>$F_T(3)$</td>
</tr>
<tr>
<td>4th eigenvector $A_Q(4)$</td>
<td>$F_Q(4)$</td>
<td>$F_T(4)$</td>
</tr>
</tbody>
</table>
**Stage 2: Salient features after PCA**

**Object:** Car

<table>
<thead>
<tr>
<th>LSK $W_Q(x_l - x; 2)$</th>
<th><strong>Query</strong> $Q$</th>
<th><strong>Target</strong> $T$</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>1st eigenvector</strong> $A_Q(1)$</td>
<td>$F_Q(1)$</td>
<td>$F_T(1)$</td>
</tr>
<tr>
<td><strong>2nd eigenvector</strong> $A_Q(2)$</td>
<td>$F_Q(2)$</td>
<td>$F_T(2)$</td>
</tr>
<tr>
<td><strong>3rd eigenvector</strong> $A_Q(3)$</td>
<td>$F_Q(3)$</td>
<td>$F_T(3)$</td>
</tr>
<tr>
<td><strong>4th eigenvector</strong> $A_Q(4)$</td>
<td>$F_Q(4)$</td>
<td>$F_T(4)$</td>
</tr>
</tbody>
</table>

**Eigenvectors** | **Query features** | **Target features**
System Overview: Stage 3

1) Resemblance Map (RM) using Matrix Cosine Similarity

RM: $f(\rho)_i$

2) Significance Tests

3) Non-maxima Suppression

Final result

stage 1

stage 2

stage 3

Compute local steering kernels

Compute feature images
Stage 3: Finding similarity between features

Target image is divided into a set of overlapping patches

\[ F_Q \leftrightarrow F_{T_i} \]
Stage 3: Correlation based Metric

The vector cosine similarity

\[ \rho(a, b) = \langle \frac{a}{\|a\|}, \frac{b}{\|b\|} \rangle = \frac{a'b}{\|a\| \|b\|} = \cos \theta \in [-1, 1], \]

Inner product between two normalized vectors

Measures angle while discarding the magnitude

Milanfar et al. EE Dept, UCSC
Stage 3: Correlation based Metric

The vector cosine similarity

\[ \rho(f_Q, f_{T_i}) = \langle \frac{f_Q}{\|f_Q\|}, \frac{f_{T_i}}{\|f_{T_i}\|} \rangle = \frac{f_Q^T f_{T_i}}{\|f_Q\| \|f_{T_i}\|} = \cos \theta_i \in [-1, 1], \]

\[ f_Q, f_{T_i} \in \mathbb{R}^d \]

Inner product between two normalized vectors

Measures angle while discarding the magnitude

Milanfar et al. EE Dept, UCSC
Stage 3: Matrix Cosine Similarity

What about a set of vectors?  Matrix Cosine Similarity

→ Frobenius Inner product between normalized matrices

\[
\rho(A, B) = \frac{\langle \overline{A}, \overline{B} \rangle_F}{\|A\|_F \|B\|_F} = \text{trace} \left( \frac{A'B}{\|A\|_F \|B\|_F} \right) \in [-1, 1],
\]

\[
= \sum_{\ell=1}^{n} \frac{a^{\ell^t}b^{\ell}}{\|A\|_F \|B\|_F},
\]

\[
= \sum_{\ell=1}^{n} \rho(a^{\ell^t}, b^{\ell}) \frac{\|a^{\ell}\| \|b^{\ell}\|}{\|A\|_F \|B\|_F}.
\]
Stage 3: Matrix Cosine Similarity

What about a set of vectors? Matrix Cosine Similarity

→ Frobenius Inner product between normalized matrices

\[
\rho(F_Q, F_{T_i}) = \frac{F_Q' F_{T_i}}{||F_Q||_F ||F_{T_i}||_F} \in [-1, 1],
\]

\[
= \sum_{\ell=1}^{n} \frac{f_Q^\ell f_{T_i}^\ell}{||F_Q||_F ||F_{T_i}||_F},
\]

\[
= \sum_{\ell=1}^{n} \rho(f_Q^\ell, f_{T_i}^\ell) \frac{||f_Q^\ell||_F ||f_{T_i}^\ell||_F}{||F_Q||_F ||F_{T_i}||_F}.
\]

A weighted sum of the column-wise vector cosine similarities

\[
= \rho(\text{colstack}(F_Q), \text{colstack}(F_{T_i}))
\]

We can prove optimality of this approach in a naïve Bayes sense.

Milanfar et al. EE Dept, UCSC
Stage 3: Generate Resemblance Map

Resemblance Map (RM)

\[ RM : f(\rho_i) = \frac{\rho_i^2}{1 - \rho_i^2} \]

Describes the proportion of variance in common between two features

Lawley-Hotelling Trace statistic

RM : \(|\rho_i|\)

RM : \(\frac{\rho_i^2}{1 - \rho_i^2}\)
Stage 3: Non-parametric Significance Tests

1. Is any sufficiently similar object present?

\[ \max f(\rho_i) > \tau_0 \]

i.e., \( \tau_0 = 0.96 \) so that \(~ 50 \% \) of variance in common

2. How many objects of interest are present?

Empirical PDF

99\% Significance level

\[ \tau \]
Experimental Results

Dataset from Weizmann Inst.

Milanfar et al. EE Dept, UCSC
Experimental Results

query

target

query

target
Experimental Results

query

target

target
Experimental Results

query

target

target

target

Milanfar et al. EE Dept, UCSC
Experimental Results

<table>
<thead>
<tr>
<th>Query</th>
<th>Target</th>
<th>Higher resemblance</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="query_image" alt="Image of query" /></td>
<td><img src="target_image" alt="Image of target" /></td>
<td>![Higher resemblance colors]</td>
</tr>
</tbody>
</table>

Degree of resemblance:
- Higher resemblance
- Lower resemblance
Experimental Results

Weizmann Inst. Object Test Set

Detection rate = TP/(TP+FN)
False positive rate = FP/(FP+TN)
Experimental Results

The MIT-CMU Face Test Set

Query

Milanfar et al. EE Dept, UCSC
Experimental Results

The MIT-CMU Face Test Set

ROC curve

Detection rate

False positive rate
Gallery Set: 10 subjects x 25 different conditions
Gallery Set: 10 subjects x 25 different conditions

Query
query

target

output

query

target

output
Outline

I. Motivation
II. System Overview
III. Object Detection
IV. Action Detection
V. Conclusion and Future work
Action Detection System Overview

1) Resemblance Map (RM) using Matrix Cosine Similarity

2) Significance Tests
3) Non-maxima Suppression

Final result

\[ \text{RM} : f(\rho)_i \]
\[ \max(f(\rho)_i) \]

- No Motion Estimation
- No Segmentation
- No Learning
- No Prior Information


Milanfar et al. EE Dept, UCSC
Stage 1: Space – Time Descriptors

\[ K(x_l - x) = \frac{\sqrt{\det(C_l)}}{2h^2} \exp\left\{ -\frac{(x_l - x)'C_l(x_l - x)}{2h^2} \right\}. \]

\[ C_l : 3x3 \text{ local covariance matrix} \]

\[ x : \text{space-time coordinates} \ [x_1, x_2, t] \]
Experimental Results

Shechtman’s action test set (Beach walk)

Query

Typical run time for target (50 frames of 144 x 192) and query (13 frames of 90 x 110): a little over 1 minute
Experimental Results
(Multiple Actions)

Multiple queries
Automatic cropping

Experimental Results
(Multiple Actions)

Multiple queries
Automatic cropping
Action Recognition

- Automatic cropping of a short action clip (25 frames)

Query

ACTION DETECTION

Scoring

\[
\begin{align*}
\text{Rank} & \quad \text{Action Category}
\end{align*}
\]

most similar

least similar

Milanfar et al. EE Dept, UCSC
**Action Classification Performance**

**Average confusion matrices**

<table>
<thead>
<tr>
<th>Classification rate: 96 %</th>
<th>Classification rate: 95.66 %</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bend</td>
<td>box</td>
</tr>
<tr>
<td>Jack</td>
<td>hclp</td>
</tr>
<tr>
<td>Jump</td>
<td>hwav</td>
</tr>
<tr>
<td>Pjump</td>
<td>jog</td>
</tr>
<tr>
<td>Run</td>
<td>run</td>
</tr>
<tr>
<td>Side</td>
<td>walk</td>
</tr>
<tr>
<td>Skip</td>
<td></td>
</tr>
<tr>
<td>Wave1</td>
<td></td>
</tr>
<tr>
<td>Wave2</td>
<td></td>
</tr>
</tbody>
</table>

(Weizmann dataset) 90 video sequences

(KTH dataset) 600 video sequences

**Classification rate** = 1 – (# of miss classification) / (total # of sequences)

**Evaluation setting:** Leave-one-out
Classify each testing video as one of the predefined classes by 3-NN (nearest neighbor)
### Action Classification Performance

**Comparison with state-of-the-art methods (KTH dataset)**

<table>
<thead>
<tr>
<th>Our Approach (1-NN)</th>
<th>89%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Our Approach (2-NN)</td>
<td>93%</td>
</tr>
<tr>
<td>Our Approach (3-NN)</td>
<td>95.66%</td>
</tr>
</tbody>
</table>

Our Approach (3-NN) | 95.66%

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Kim et al. (2008)</td>
<td>95.33%</td>
</tr>
<tr>
<td>Ali et al. (2008)</td>
<td>87.7%</td>
</tr>
<tr>
<td>Dollar et al. (2005)</td>
<td>81.17%</td>
</tr>
<tr>
<td>Ning et al. (2008)</td>
<td>92.31%</td>
</tr>
<tr>
<td>Niebles et al. (2008)</td>
<td>81.5%</td>
</tr>
<tr>
<td>Wong et al. (2007)</td>
<td>71.16%</td>
</tr>
</tbody>
</table>

Classification rate = $1 - \frac{\text{# of miss classification}}{\text{total # of sequences}}$

- **We outperform all the state-of-the-art methods on KTH dataset.**
Publications


• H. Seo and P. Milanfar, “Detection of Human Actions from a Single Example”, Accepted for publication in International Conference on Computer Vision (ICCV), March 2009


Conclusions & Future Work

- Local Steering Kernels are Very **Effective Descriptors**

- **Simple Approach:** PCA + Matrix Cosine Similarity

- **Excellent Detection and Recognition is Achieved without Training**

- Make algorithm **scalable for image and (video) retrieval**

- Increase accuracy by incorporating “**context**”

- Detect /recognize objects of interest in general **degraded data without explicit restoration**