Robust Image Classification Using Discriminative Graphical Models

Umamahesh Srinivas

iPAL Group Meeting

February 17, 2012
Introduction

- View image classification as a hypothesis testing problem:

\[ H_0 : x \sim f(x|H_0) \]
\[ H_1 : x \sim f(x|H_1). \]

Likelihood ratio test (LRT):\(^{1}\)

\[ L(x) := \frac{f(x|H_1)}{f(x|H_0)} \overset{H_1}{\sim} \frac{H_1}{H_0} \tau. \]

Figure: Fingerprint verification (biometrics).

- Success of Bayesian classifiers dictated by accuracy of estimation of conditional densities.
Introduction

- View image classification as a hypothesis testing problem:

\[ H_0 : \ x \sim f(x|H_0) \]
\[ H_1 : \ x \sim f(x|H_1). \]

Likelihood ratio test (LRT):

\[ L(x) := \frac{f(x|H_1)}{f(x|H_0)} \xrightarrow{H_1 \geq H_0} \tau. \]

Figure: Fingerprint verification (biometrics).

- Success of Bayesian classifiers dictated by accuracy of estimation of conditional densities.
Image classification

Two stages in any classification framework:

1. Feature extraction from acquired imagery
2. Decision engine which performs class assignment

Algorithmic developments:

- Feature sets
  - Template-based
  - Transform domain-based (e.g. wavelets)
  - Computer vision-based
  - Estimation-theoretic

- Decision engines
  - Neural networks
  - Support vector machines (SVM)
  - Boosting

- Classifier fusion: heuristic, meta-classification
  - Outputs of individual classifiers → high-level features

---

1 Srinivas et al., IEEE Radar Conference, 2011
Image classification

Two stages in any classification framework:

1. Feature extraction from acquired imagery
2. Decision engine which performs class assignment

Algorithmic developments:

• Feature sets
  • Template-based
  • Transform domain-based (e.g. wavelets)
  • Computer vision-based
  • Estimation-theoretic

• Decision engines
  • Neural networks
  • Support vector machines (SVM)
  • Boosting

• Classifier fusion: heuristic, meta-classification
  • Outputs of individual classifiers → high-level features

---

1 Srinivas et al., IEEE Radar Conference, 2011
Image classification

Two stages in any classification framework:

1. **Feature extraction** from acquired imagery
2. **Decision engine** which performs class assignment

Algorithmic developments:

- **Feature sets**
  - Template-based
  - Transform domain-based (e.g. wavelets)
  - Computer vision-based
  - Estimation-theoretic

- **Decision engines**
  - Neural networks
  - Support vector machines (SVM)
  - Boosting

- **Classifier fusion**: heuristic, meta-classification\(^1\)
  - Outputs of individual classifiers \(\rightarrow\) high-level features

---

\(^1\) Srinivas et al., IEEE Radar Conference, 2011
Image classification

Two stages in any classification framework:
1. Feature extraction from acquired imagery
2. Decision engine which performs class assignment

Algorithmic developments:

- **Feature sets**
  - Template-based
  - Transform domain-based (e.g. wavelets)
  - Computer vision-based
  - Estimation-theoretic

- **Decision engines**
  - Neural networks
  - Support vector machines (SVM)
  - Boosting

- **Classifier fusion**: heuristic, meta-classification\(^1\)
  - Outputs of individual classifiers $\rightarrow$ **high-level** features

---

\(^1\)Srinivas et al., IEEE Radar Conference, 2011
Research challenges

- Limited availability of training → serious practical concern
  - High-dimensional image data/ equivalent features

- Variety of features and decision engines
  - No single optimal feature set-decision engine combination

Motivation for contribution:

- Presence of complementary yet correlated information

- Probabilistic graphical models: learn tractable models from high-D data under limited training.
Research challenges

- Limited availability of training → serious practical concern
  - High-dimensional image data/ equivalent features

- Variety of features and decision engines
  - No single optimal feature set-decision engine combination

Motivation for contribution:

- Presence of complementary yet correlated information

- Probabilistic graphical models: learn tractable models from high-D data under limited training.
Application I: Face Recognition\textsuperscript{2,3,4}

\textsuperscript{2} In collaboration with Prof. Trac Tran, The Johns Hopkins University
\textsuperscript{3} Srinivas et al., IEEE Asilomar Conf., Nov. 2011
\textsuperscript{4} Chen et al., submitted to IEEE Trans. Image Processing, Nov. 2011
Face recognition: Overview

Problem formulation:

- $K$ unique faces (persons)
- Training: $\{v_{1,1}, \ldots, v_{1,N_1}\}, \ldots, \{v_{K,1}, \ldots, v_{K,N_k}\}$
- Goal: Given new face $y$, assign one of the labels $\{1, \ldots, K\}$

Applications: Security, biometrics, online image search, etc.

Feature extraction for dimensionality reduction:

- Eigenfaces\(^5\)
- Fisherfaces\(^6\)

Classifier (decision engine):

- Nearest neighbor, nearest subspace\(^7\)
- Support vector machines\(^8\)

\(^6\) Belhumeur et al., IEEE Trans. PAMI, 1997
\(^7\) Kriegman et al., CVPR 2003
\(^8\) Vapnik, 1995
Face recognition: Overview

Problem formulation:
- \( K \) unique faces (persons)
- Training: \( \{\mathbf{v}_{1,1}, \ldots, \mathbf{v}_{1,N_1}\}, \ldots, \{\mathbf{v}_{K,1}, \ldots, \mathbf{v}_{K,N_k}\} \)
- Goal: Given new face \( \mathbf{y} \), assign one of the labels \( \{1, \ldots, K\} \)

Applications: Security, biometrics, online image search, etc.

Feature extraction for dimensionality reduction:
- Eigenfaces
- Fisherfaces

Classifier (decision engine):
- Nearest neighbor, nearest subspace
- Support vector machines

---

7. Kriegman et al., CVPR 2003
8. Vapnik, 1995
Face recognition: Overview

Problem formulation:
- $K$ unique faces (persons)
- Training: $\{v_{1,1}, \ldots, v_{1,N_1}\}, \ldots, \{v_{K,1}, \ldots, v_{K,N_k}\}$
- Goal: Given new face $y$, assign one of the labels $\{1, \ldots, K\}$

Applications: Security, biometrics, online image search, etc.

Feature extraction for dimensionality reduction:
- Eigenfaces
- Fisherfaces

Classifier (decision engine):
- Nearest neighbor, nearest subspace
- Support vector machines

---

6 Belhumeur et al., IEEE Trans. PAMI, 1997
7 Kriegman et al., CVPR 2003
8 Vapnik, 1995
Sparse representation for face recognition\textsuperscript{9}

- **Assumption:** New face of person $i$ lies in linear span of training samples associated with class $i$

$$y = \alpha_{i,1}v_{i,1} + \alpha_{i,2}v_{i,2} + \ldots + \alpha_{i,N_i}v_{i,N_i} = A_i\alpha_i$$

$$y \in \mathbb{R}^n, A_i \in \mathbb{R}^{n \times N_i}, \alpha_i \in \mathbb{R}^{N_i}$$

- $y \rightarrow$ sparse linear combination of all training samples:

$$y = \begin{bmatrix} A_1 & A_2 & \ldots & A_K \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_K \end{bmatrix} = A\alpha$$

$$A \in \mathbb{R}^{n \times T}, T = \sum_{i=1}^{K} N_i, \alpha \in \mathbb{R}^T$$

- Membership of $y$ encoded by sparse representation

$$\alpha = [0^t \ldots 0^t \alpha_i^t \ 0^t \ldots 0^t]^t.$$
Sparse representation for face recognition

- **Assumption**: New face of person $i$ lies in linear span of training samples associated with class $i$

  $$y = \alpha_{i,1}v_{i,1} + \alpha_{i,2}v_{i,2} + \ldots + \alpha_{i,N_i}v_{i,N_i} = A_i\alpha_i$$

  \( (y \in \mathbb{R}^n, A_i \in \mathbb{R}^{n \times N_i}, \alpha_i \in \mathbb{R}^{N_i}) \)

- $y$ → **sparse** linear combination of all training samples:

  $$y = \begin{bmatrix} A_1 & A_2 & \ldots & A_K \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_K \end{bmatrix} = A\alpha$$

  \( (A \in \mathbb{R}^{n \times T}, T = \sum_{i=1}^{K} N_i, \alpha \in \mathbb{R}^{T}) \)

- Membership of $y$ encoded by sparse representation

  $$\alpha = [0^t \ldots 0^t \alpha_i^t 0^t \ldots 0^t]^t.$$
Sparse representation for face recognition

- **Assumption:** New face of person $i$ lies in linear span of training samples associated with class $i$

$$y = \alpha_{i,1}v_{i,1} + \alpha_{i,2}v_{i,2} + \ldots + \alpha_{i,N_i}v_{i,N_i} = A_i\alpha_i$$

($y \in \mathbb{R}^n, A_i \in \mathbb{R}^{n \times N_i}, \alpha_i \in \mathbb{R}^{N_i}$)

- $y \rightarrow$ **sparse** linear combination of all training samples:

$$y = \begin{bmatrix} A_1 & A_2 & \ldots & A_K \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_K \end{bmatrix} = A\alpha$$

($A \in \mathbb{R}^{n \times T}, T = \sum_{i=1}^{K} N_i, \alpha \in \mathbb{R}^T$)

- Membership of $y$ encoded by sparse representation

$$\alpha = \begin{bmatrix} 0^t & \ldots & 0^t & \alpha_i^t & 0^t & \ldots & 0^t \end{bmatrix}^t.$$
Sparse representation for face recognition (contd.)

- Solve the sparse recovery problem:

\[ \hat{\alpha} = \arg \min \|\alpha\|_0 \quad \text{subject to} \quad \|A\alpha - y\|_2 \leq \epsilon \]

Convex relaxation (if solution is sparse enough):

\[ \hat{\alpha} = \arg \min \|\alpha\|_1 \quad \text{subject to} \quad \|A\alpha - y\|_2 \leq \epsilon \]

- Class decision based on reconstruction residuals:

\[ \text{identity}(y) = \arg \min_i \|y - A\delta_i(\hat{\alpha})\|_2 \]

\[ \delta_i(\hat{\alpha}) \rightarrow \text{only non-zero entries are those associated with class } i \]

- Robustness to variety of distortions.
Sparse representation for face recognition (contd.)

- Solve the sparse recovery problem:
  \[
  \hat{\alpha} = \arg \min_\alpha \|\alpha\|_0 \quad \text{subject to} \quad \|A\alpha - y\|_2 \leq \epsilon
  \]

  Convex relaxation (if solution is sparse enough):
  \[
  \hat{\alpha} = \arg \min_\alpha \|\alpha\|_1 \quad \text{subject to} \quad \|A\alpha - y\|_2 \leq \epsilon
  \]

- Class decision based on reconstruction residuals:
  \[
  \text{identity}(y) = \arg \min_i \|y - A\delta_i(\hat{\alpha})\|_2
  \]

  \(\delta_i(\hat{\alpha}) \rightarrow\) only non-zero entries are those associated with class \(i\)

- Robustness to variety of distortions.
Sparse representation for face recognition (contd.)

- Solve the sparse recovery problem:

\[ \hat{\alpha} = \arg \min_{\alpha} \|\alpha\|_0 \quad \text{subject to} \quad \|A\alpha - y\|_2 \leq \epsilon \]

Convex relaxation (if solution is sparse enough):

\[ \hat{\alpha} = \arg \min_{\alpha} \|\alpha\|_1 \quad \text{subject to} \quad \|A\alpha - y\|_2 \leq \epsilon \]

- Class decision based on reconstruction residuals:

\[ \text{identity}(y) = \arg \min_i \|y - A\delta_i(\hat{\alpha})\|_2 \]

\[ \delta_i(\hat{\alpha}) \rightarrow \text{only non-zero entries are those associated with class } i \]

- Robustness to variety of distortions.
Figure: Left: Varying amounts of random pixel corruption. Right: Recognition rate variation with corruption.
Drawbacks and challenges

1. Accurate registration of training and test images necessary
   - Misalignment: translation, rotation, scale; pose and illumination variation; occlusion
   - Computational cost and feasibility in practical recognition systems

2. Class decision using reconstruction residuals
   - Does not capture inter-class variation
   - Sparse representations inherently discriminative.
Local features for recognition: Motivation
Local features for recognition: Motivation
Local features for recognition: Motivation
Local features for recognition: Motivation
Local features for recognition: Motivation
Local features for recognition: Motivation
Local features for recognition: Motivation
Local features for recognition: Motivation
Local sparsity model for robust face recognition\textsuperscript{10}

- Inspired by block-based motion estimation
- Block-sparsity model using \textit{locally adaptive} dictionary $D_{ij}$
- No explicit estimation of registration parameters.

\textsuperscript{10}Chen et al., IEEE ICIP 2010
How to build the dictionary?
Block sparsity for face recognition

- For block $y_{ij}$ in misaligned test image $Y$,

$$
\hat{\alpha}_{ij} = \arg \min \| \alpha_{ij} \|_0 \quad \text{subject to} \quad \| D_{ij} \alpha_{ij} - y_{ij} \|_2 \leq \epsilon
$$

- Identity of block $y_{ij}$: determined by the residuals

$$
\text{identity} (y_{ij}) = \arg \min_{k=1,\ldots,K} r_{ij}^k,
$$

$$
r_{ij}^k = \| y_{ij} - D_{ij} \hat{\alpha}_{ij}^k \|_2
$$

- Select multiple local blocks from image → obtain individual classification decisions

- How to combine local decisions into global class decision?

  - Voting and ensemble classifiers

  - Challenge: Principled strategy to combine correlated sparse representations.
Block sparsity for face recognition

- For block $y_{ij}$ in misaligned test image $Y$,
  \[
  \hat{\alpha}_{ij} = \arg \min_{\alpha_{ij}} \|\alpha_{ij}\|_0 \quad \text{subject to} \quad \|D_{ij} \alpha_{ij} - y_{ij}\|_2 \leq \epsilon
  \]

- Identity of block $y_{ij}$: determined by the residuals
  \[
  \text{identity}(y_{ij}) = \arg \min_{k=1,\ldots,K} r_{ij}^k,
  \]
  \[
  r_{ij}^k = \|y_{ij} - D_{ij} \hat{\alpha}_{ij}^k\|_2
  \]

- Select multiple local blocks from image $\rightarrow$ obtain individual classification decisions

- How to combine local decisions into global class decision?
  - Voting and ensemble classifiers
  - Challenge: Principled strategy to combine correlated sparse representations.
Block sparsity for face recognition

- For block $y_{ij}$ in misaligned test image $Y$,
  \[
  \hat{\alpha}_{ij} = \arg \min \|\alpha_{ij}\|_0 \quad \text{subject to} \quad \|D_{ij}\alpha_{ij} - y_{ij}\|_2 \leq \epsilon
  \]

- Identity of block $y_{ij}$: determined by the residuals
  \[
  \text{identity}(y_{ij}) = \arg \min_{k=1,\ldots,K} r_{ij}^k,
  \]
  \[
  r_{ij}^k = \|y_{ij} - D_{ij}\hat{\alpha}_{ij}^k\|_2
  \]

- Select multiple local blocks from image $\rightarrow$ obtain individual classification decisions

- How to combine local decisions into global class decision?
  - Voting and ensemble classifiers
  - **Challenge:** Principled strategy to combine correlated sparse representations.
Probabilistic graphical models: A brief review

- **Graph** $G = (\mathcal{V}, \mathcal{E})$ defined by a set of nodes $\mathcal{V} = \{1, \ldots, n\}$, and a set of edges $\mathcal{E} \subset \binom{\mathcal{V}}{2}$.

- **Graphical model**: Random vector defined on a graph; nodes represent random variables, edges reveal conditional dependencies.

- **Graph structure defines factorization of joint probability distribution**

  $$f(\mathbf{x}) = f(x_1)f(x_2|x_1)f(x_3|x_1)f(x_4|x_2)f(x_5|x_2)f(x_6|x_3)f(x_7|x_3).$$

**Figure**: Tree - connected acyclic graph.
Learning graphical models

- **Generative learning**: Single graph to minimize approximation error

  Given \( p \), find \( \hat{p} = \arg \min_{\hat{p}} D(p||\hat{p}) \).

  \[
  D(p||\hat{p}) := \int p(x) \log \left( \frac{p(x)}{\hat{p}(x)} \right) dx \rightarrow \text{KL-divergence.}
  \]

- **Discriminative learning**: Simultaneously learn a pair of graphs to approximately minimize classification error

  \( \hat{p}, \hat{q} = \arg \max_{\hat{p} \in T} \hat{J}(\hat{p}, \hat{q}; p, q) \).

---

Learning graphical models

- **Generative learning**: Single graph to minimize approximation error

  Given $p$, find $\hat{p} = \arg\min_{\hat{p} \text{ is a tree}} D(p||\hat{p})$.

  \[
  \left( D(p||\hat{p}) := \int p(x) \log \left( \frac{p(x)}{\hat{p}(x)} \right) dx \rightarrow \text{KL-divergence}. \right)
  \]

- **Discriminative learning**: Simultaneously learn a pair of graphs to approximately minimize classification error

  Tree-approximate $J$-divergence:

  \[
  \hat{J}(\hat{p}, \hat{q}; p, q) := \int_{\Omega \subset \mathcal{X}^n} (p(x) - q(x)) \log \left( \frac{\hat{p}(x)}{\hat{q}(x)} \right) dx.
  \]

  \[
  (\hat{p}, \hat{q}) = \arg\max_{\hat{p} \in \mathcal{T}_p, \hat{q} \in \mathcal{T}_q} \hat{J}(\hat{p}, \hat{q}; p, q).
  \]

---

Discriminative graphical models for classification\textsuperscript{13}

Two-stage framework:

1. Acquire multiple signal representations, which are \textit{conditionally correlated} per class

2. Mine dependencies between different features via boosting on discriminative graphs.

\textsuperscript{13} Srinivas et al., IEEE ICIP, Sep. 2011
Contribution: Robust face recognition using discriminative graphical models

Figure: Learning graphs on sparse features.
Learning discriminative graphs: An illustration

Iteration 1:

(a) Initial graph
(b) Newly-learned tree
(c) Augmented graph

Re-weighting of training samples (boosting) → learn another tree ...

14 Shown for distribution $p$; graph for $q$ learnt analogously.
Learning discriminative graphs: An illustration

Iteration 1:

(a) Initial graph  (b) Newly-learned tree  (c) Augmented graph

Re-weighting of training samples (boosting) $\rightarrow$ learn another tree . . .

$^{14}$ Shown for distribution $p$; graph for $q$ learnt analogously.
Learning discriminative graphs: An illustration

Iteration 2:

(a) Initial graph  (b) Newly-learned tree  (c) Augmented graph

Newly introduced edges crucial for capturing correlations amongst distinct signal representations.
Learning discriminative graphs: An illustration

Iteration 3:

(a) Initial graph
(b) Newly-learned tree
(c) Augmented graph
Learning discriminative graphs: An illustration

Iteration 4:

(a) Initial graph
(b) Newly-learned tree
(c) Augmented graph
Stopping criterion

How many edges to learn?

1. Cross-validation

2. Using the J-divergence:

\[ \hat{J}(\hat{p}, \hat{q}; p, q) := \int_{\Omega \subseteq \mathcal{X}^n} (p(x) - q(x)) \log \left( \frac{p(x)}{\hat{q}(x)} \right) dx. \]

Stopping criterion:
Stop after \( i \) boosting iterations if:

\[ \frac{\hat{J}^{(i+1)}(\hat{p}, \hat{q}; p, q) - \hat{J}^{(i)}(\hat{p}, \hat{q}; p, q)}{\hat{J}^{(i)}(\hat{p}, \hat{q}; p, q)} < \epsilon \]
Learning thicker graphical models

- Final boosted classifier:

\[
H_T(x) = \text{sgn} \left[ \sum_{t=1}^{T} \alpha_t \log \left( \frac{\hat{p}_t(x)}{\hat{q}_t(x)} \right) \right] = \text{sgn} \left[ \log \prod_{t=1}^{T} \left( \frac{\hat{p}_t(x)}{\hat{q}_t(x)} \right)^{\alpha_t} \right]
\]

\[
= \text{sgn} \left[ \log \left( \frac{\prod_{t=1}^{T} \hat{p}_t(x)^{\alpha_t}}{\prod_{t=1}^{T} \hat{q}_t(x)^{\alpha_t}} \right) \right] = \text{sgn} \left[ \log \left( \frac{\hat{p}(x)}{\hat{q}(x)} \right) \right]
\]

Define:

\[
Z_p(\alpha) = Z_p(\alpha_1, \ldots, \alpha_T) = \sum_x \hat{p}(x); \quad Z_q(\alpha) = \sum_x \hat{q}(x)
\]

- Normalized distributions for inference: \( \frac{\hat{p}(x)}{Z_p(\alpha)}, \frac{\hat{q}(x)}{Z_q(\alpha)} \)

→ Thicker graphical models learnt.
Robust face recognition using graphical models

\[ i^* = \arg \max_{i \in \{1, \ldots, K\}} \log \left( \frac{\hat{f}_p^i(\alpha)}{\hat{f}_q^i(\alpha)} \right). \] (1)

**Algorithm 1** Local-Sparse-Graphical-Model (LSGM) (Steps 1-4 offline)

1. **Feature extraction (training):** Obtain sparse representations \( \alpha_l, l = 1, \ldots, P \) in \( \mathbb{R}^m \) from facial features, using local block-sparsity model
2. **Initial disjoint graphs:**
   For \( l = 1, \ldots, P \)
   Discriminatively learn pairs of \( m \)-node tree graphs \( G^p_l \) and \( G^q_l \) on \( \{\alpha_l\} \)
3. Separately concatenate nodes corresponding to \( p \) and \( q \) respectively
4. **Boosting on disjoint graphs:** Iteratively thicken initial disjoint graphs via boosting to obtain final graphs \( G^p \) and \( G^q \)

{**Online process**}

5. **Feature extraction (test):** Obtain sparse representations \( \alpha_l, l = 1, \ldots, P \) in \( \mathbb{R}^m \) from test image
6. **Inference:** Classify based on output of the resulting classifier using (1).
Results: Rotation (Extended Yale B)

- SRC: sparse representation-based classification
- LSGM: local sparsity with graphical models.
Results: Scaling (Yale)

Table: Recognition rate using SRC and LSGM.

<table>
<thead>
<tr>
<th>SF</th>
<th>1</th>
<th>1.071</th>
<th>1.143</th>
<th>1.214</th>
<th>1.286</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>100</td>
<td>100</td>
<td>98.0</td>
<td>88.2</td>
<td>76.5</td>
</tr>
<tr>
<td></td>
<td>98.8</td>
<td>98.2</td>
<td>98.5</td>
<td>97.5</td>
<td>97.5</td>
</tr>
<tr>
<td>1.063</td>
<td>99.7</td>
<td>96.5</td>
<td>86.1</td>
<td>68.5</td>
<td>50.3</td>
</tr>
<tr>
<td></td>
<td>97.5</td>
<td>96.7</td>
<td>96.0</td>
<td>96.0</td>
<td>93.5</td>
</tr>
<tr>
<td>1.125</td>
<td>83.8</td>
<td>70.2</td>
<td>49.8</td>
<td>33.6</td>
<td>26.2</td>
</tr>
<tr>
<td></td>
<td>97.4</td>
<td>96.5</td>
<td>96.2</td>
<td>95.2</td>
<td>93.2</td>
</tr>
<tr>
<td>1.188</td>
<td>54.5</td>
<td>43.7</td>
<td>26.8</td>
<td>20.0</td>
<td>18.0</td>
</tr>
<tr>
<td></td>
<td>94.9</td>
<td>92.9</td>
<td>91.6</td>
<td>89.4</td>
<td>87.1</td>
</tr>
<tr>
<td>1.25</td>
<td>36.1</td>
<td>27.2</td>
<td>20.9</td>
<td>16.6</td>
<td>12.3</td>
</tr>
<tr>
<td></td>
<td>94.9</td>
<td>93.0</td>
<td>92.2</td>
<td>87.9</td>
<td>82.0</td>
</tr>
<tr>
<td>1.313</td>
<td>31.5</td>
<td>24.3</td>
<td>16.7</td>
<td>13.9</td>
<td>10.6</td>
</tr>
<tr>
<td></td>
<td>90.7</td>
<td>90.4</td>
<td>84.1</td>
<td>81.0</td>
<td>75.5</td>
</tr>
</tbody>
</table>

Table: Overall recognition rates.

<table>
<thead>
<tr>
<th>Method</th>
<th>Recog. rate (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>LSGM</td>
<td>89.4</td>
</tr>
<tr>
<td>SRC</td>
<td>60.8</td>
</tr>
<tr>
<td>Eigen-NS</td>
<td>55.5</td>
</tr>
<tr>
<td>Eigen-SVM</td>
<td>56.7</td>
</tr>
<tr>
<td>Fisher-NS</td>
<td>54.1</td>
</tr>
<tr>
<td>Fisher-SVM</td>
<td>57.1</td>
</tr>
</tbody>
</table>
Results: Scaling and random pixel corruption (Yale)

Table: Test images scaled and subjected to random pixel corruption.

<table>
<thead>
<tr>
<th>Method</th>
<th>Recognition rate (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>LSGM</td>
<td>96.3</td>
</tr>
<tr>
<td>SRC</td>
<td>93.2</td>
</tr>
<tr>
<td>Eigen-NS</td>
<td>54.3</td>
</tr>
<tr>
<td>Eigen-SVM</td>
<td>58.5</td>
</tr>
<tr>
<td>Fisher-NS</td>
<td>56.2</td>
</tr>
<tr>
<td>Fisher-SVM</td>
<td>59.9</td>
</tr>
</tbody>
</table>
Results: Scaling and disguise (AR database)

Table: Test images scaled and subjects wear disguise.

<table>
<thead>
<tr>
<th>Method</th>
<th>Recog. rate (%)</th>
<th>Recog. rate (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Sunglasses</td>
<td>Scarves</td>
</tr>
<tr>
<td>LSGM</td>
<td>96.0</td>
<td>92.9</td>
</tr>
<tr>
<td>SRC</td>
<td>93.5</td>
<td>90.1</td>
</tr>
<tr>
<td>Eigen-NS</td>
<td>47.2</td>
<td>29.6</td>
</tr>
<tr>
<td>Eigen-SVM</td>
<td>53.5</td>
<td>34.5</td>
</tr>
<tr>
<td>Fisher-NS</td>
<td>57.9</td>
<td>41.7</td>
</tr>
<tr>
<td>Fisher-SVM</td>
<td>61.7</td>
<td>43.6</td>
</tr>
</tbody>
</table>
Results: Outlier rejection

ROC for SRC:

SCI \( (\alpha) = K \cdot \max_i \| \delta_i(\alpha) \|_1 \| \alpha \|_1 - 1 \)

\( K \in [0, 1] \).

Figure: ROC curves for outlier rejection.
Results: Outlier rejection

- Subset of total number of classes for training
- Test images rotated by 5 degrees
- ROC for SRC:

\[
\text{SCI}(\alpha) = \frac{K \cdot \max_i \frac{\|\delta_i(\alpha)\|_1}{\|\alpha\|_1} - 1}{K - 1} \in [0, 1].
\]
Results: Outlier rejection

- Subset of total number of classes for training
- Test images rotated by 5 degrees
- ROC for SRC:

\[ \text{SCI}(\alpha) = \frac{K \cdot \max_i \frac{\| \delta_i(\alpha) \|_1}{\| \alpha \|_1} - 1}{K - 1} \in [0, 1]. \]

Figure: ROC curves for outlier rejection.
Application II: Automatic Target Recognition$^{15,16,17}$

15 Joint work with Dr. Raghu Raj, NRL
16 Srinivas et al., IEEE ICIP, Sep. 2011
Automatic Target Recognition: Review

- Exploit imagery from diverse sensed sources for automatic target identification

- **Sources**: Synthetic aperture radar (SAR), inverse SAR, infra-red (FLIR), hyperspectral, etc.

**Figure**: Schematic of ATR framework. The classification and recognition stages assign an input image/feature to one of many target classes.
Learning Discriminative Graphical Models for ATR

Two-stage framework:

1. Acquire multiple signal representations, which are conditionally correlated per class

2. Mine dependencies between different features via boosting on discriminative graphs.
Stage 1: Feature extraction

- Projection to a lower-dimensional space $\mathcal{P} : \mathbb{R}^n \mapsto \mathbb{R}^m$, $m < n$

- $M$ different projections $\mathcal{P}_i$, $i = 1, \ldots, M$, generate corresponding low-level features $\mathbf{y}_i \in \mathbb{R}^{m_i}$

---

18 For notational simplicity, we let $m_1 = m_2 = \ldots = m$. 

02/17/2012 iPAL Group Meeting
Stage 2: Learning discriminative graphs

Boosting on initially disjoint graphs to discover new edges (conditional correlations)
What about signal representations?

- **Blind** discriminative learning: no prior information about images
- Projection to wavelet sub-bands\textsuperscript{19,20,21}
  - 2-D Reverse biorthogonal wavelets

![Image](image_url)

**Figure**: LL sub-band, LH sub-band, HL sub-band.

\textsuperscript{19} Fukuda et al., IEEE Trans. Geoscience and Remote Sensing, 1999
\textsuperscript{20} Simard et al., IEEE IGARSS, 1999
\textsuperscript{21} N. Sandirasegaram, Tech. Memo. DRDC Ottawa, 2005
Experiment: Multi-class classification for ATR

Five classes from benchmark MSTAR database:

1. T-72 tanks
2. BMP-2 infantry fighting vehicles
3. BTR-70 armored personnel carriers
4. ZIL131 trucks
5. D7 tractors

- Processed input image dimension - $64 \times 64$
- **Training:** 150 images per class; **testing:** 1913 images
- Compare with single feature set + SVM.

---

22 Extension of binary classification in one-versus-all manner.
Experiment: Multi-class classification for ATR

Using wavelet basis representations:

Table: Confusion matrix for LL wavelet sub-band feature + SVM.

<table>
<thead>
<tr>
<th>Class</th>
<th>BMP-2</th>
<th>BTR-70</th>
<th>T-72</th>
<th>ZIL131</th>
<th>D7</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMP-2</td>
<td>0.85</td>
<td>0.04</td>
<td>0.04</td>
<td>0.03</td>
<td>0.04</td>
</tr>
<tr>
<td>BTR-70</td>
<td>0.05</td>
<td>0.87</td>
<td>0.03</td>
<td>0.02</td>
<td>0.03</td>
</tr>
<tr>
<td>T-72</td>
<td>0.04</td>
<td>0.07</td>
<td>0.86</td>
<td>0.01</td>
<td>0.02</td>
</tr>
<tr>
<td>ZIL131</td>
<td>0.01</td>
<td>0.05</td>
<td>0.06</td>
<td>0.85</td>
<td>0.03</td>
</tr>
<tr>
<td>D7</td>
<td>0.04</td>
<td>0.0</td>
<td>0.06</td>
<td>0.06</td>
<td>0.84</td>
</tr>
</tbody>
</table>

Table: Confusion matrix for proposed approach using wavelet basis.

<table>
<thead>
<tr>
<th>Class</th>
<th>BMP-2</th>
<th>BTR-70</th>
<th>T-72</th>
<th>ZIL131</th>
<th>D7</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMP-2</td>
<td>0.92</td>
<td>0.05</td>
<td>0.02</td>
<td>0.01</td>
<td>0.01</td>
</tr>
<tr>
<td>BTR-70</td>
<td>0.03</td>
<td>0.94</td>
<td>0.02</td>
<td>0.0</td>
<td>0.01</td>
</tr>
<tr>
<td>T-72</td>
<td>0.02</td>
<td>0.05</td>
<td>0.91</td>
<td>0.0</td>
<td>0.02</td>
</tr>
<tr>
<td>ZIL131</td>
<td>0.01</td>
<td>0.02</td>
<td>0.03</td>
<td>0.93</td>
<td>0.01</td>
</tr>
<tr>
<td>D7</td>
<td>0.01</td>
<td>0.0</td>
<td>0.04</td>
<td>0.04</td>
<td>0.91</td>
</tr>
</tbody>
</table>
Experiment: Performance as function of training size

- Practical concern for ATR: limited training resources
- Binary classification problem: T-72 and BMP-2 classes
- Probability of misclassification $\rightarrow$ average of false-alarm and miss probabilities.

---

23 Rizvi et al., Applied Imagery Pattern Recognition Workshop, 2003
Experiment: Performance as function of training size

- Practical concern for ATR: limited training resources
- Binary classification problem: T-72 and BMP-2 classes
- Probability of misclassification → average of false-alarm and miss probabilities.
- Five approaches compared:
  1. **IndSVM**: single feature set + SVM
  2. **ClassFusion**: ranking-based classifier fusion
  3. **AdaBoost**: boosting-based approach
  4. **CombSVM**: concatenated feature vector + SVM
  5. **IGT**: Proposed iterative graph thickening framework

---

23 Rizvi et al., Applied Imagery Pattern Recognition Workshop, 2003

02/17/2012 iPAL Group Meeting
Locality-based discriminative learning

(a) Optical image.  (b) SAR image.

- Local image features more useful than global features
- Exploit scene-specific structure via image segmentation
- Wavelet LL sub-band from each region as feature.
Results: Wavelet basis

Figure: Classification error vs. training sample size. Individual feature dimension $m = 64$ (except for the local IGT method).
Conclusions

- Developed a probabilistic graphical model framework to mine conditional dependencies between distinct sets of image features for classification tasks

- Sub-optimal discriminative graphs learnt are particularly meritorious in the difficult regime of low training, high dimensionality

- Application to robust face recognition: Local block-based sparsity model → robustness to alignment errors and variety of distortions
Backup Slides
Edge weights:

\[
\begin{align*}
\psi^p_{i,j} &:= \mathbb{E}_{\tilde{p}_{i,j}} \left[ \log \frac{\tilde{p}_{i,j}}{\tilde{p}_i \tilde{p}_j} \right] - \mathbb{E}_{\tilde{q}_{i,j}} \left[ \log \frac{\tilde{q}_{i,j}}{\tilde{q}_i \tilde{q}_j} \right] \\
\psi^q_{i,j} &:= \mathbb{E}_{\tilde{q}_{i,j}} \left[ \log \frac{\tilde{q}_{i,j}}{\tilde{q}_i \tilde{q}_j} \right] - \mathbb{E}_{\tilde{p}_{i,j}} \left[ \log \frac{\tilde{p}_{i,j}}{\tilde{p}_i \tilde{p}_j} \right].
\end{align*}
\]

**Algorithm 2** Discriminative trees (DT)

Given: Training sets \( \mathcal{T}_p \) and \( \mathcal{T}_q \).

1. Estimate pairwise statistics \( \tilde{p}_{i,j}(x_i, x_j), \tilde{q}_{i,j}(x_i, x_j) \) for all edges \((i, j)\).
2. Compute edge weights \( \psi^p_{i,j} \) and \( \psi^q_{i,j} \) for all edges \((i, j)\).
3. Find \( \mathcal{E}_{\tilde{p}} = \text{MWST}(\psi^p_{i,j}) \) and \( \mathcal{E}_{\tilde{q}} = \text{MWST}(\psi^q_{i,j}) \).
4. Get \( \text{\hat{p}} \) by projection of \( \tilde{p} \) onto \( \mathcal{E}_{\tilde{p}} \); likewise \( \text{\hat{q}} \).
5. LRT using \( \text{\hat{p}} \) and \( \text{\hat{q}} \).
### Algorithm 3 AdaBoost learning algorithm

1: **Input data** \((x_i, y_i), i = 1, 2, \ldots, N\), where \(x_i \in S, y_i \in \{-1, +1\}\)
2: **Initialize** \(D_1(i) = \frac{1}{N}, i = 1, 2, \ldots, N\)
3: **For** \(t = 1, 2, \ldots, T\):
   - Train weak learner using distribution \(D_t\)
   - Determine weak hypothesis \(h_t : S \mapsto \mathbb{R}\) with error \(\epsilon_t\)
   - Choose \(\beta_t = \frac{1}{2} \ln \left( \frac{1 - \epsilon_t}{\epsilon_t} \right)\)
   - \(D_{t+1}(i) = \frac{1}{Z_t} \{D_t(i) \exp(-\beta_t y_i h_t(x_i))\}\), where \(Z_t\) is a normalization factor
4: **Output** soft decision \(H(x) = \text{sign} \left[ \sum_{t=1}^T \beta_t h_t(x) \right] \).

- Distribution of weights over the training set
- In each iteration, weak learner \(h_t\) minimizes weighted training error
- Weights on incorrectly classified samples increased \(\rightarrow\) slow learners penalized for harder examples.