The Open Set Recognition Problem
and Its Implications and Opportunities in Visual Computing, Forensics and Security

Anderson Rocha
University of Campinas, Brazil
anderson.rocha@ic.unicamp.br

Walter J. Scheirer
University of Notre Dame, U.S.A.
walter.scheirer@nd.edu
About me

- Associate Professor at the University of Campinas
  - Microsoft Research Faculty Fellow
  - Brazilian Academy of Sciences
  - Brazilian Academy of Forensic Sciences
- Research in Digital Forensics, Reasoning for Complex Data and Machine Intelligence

Digital Forensics  |  Reasoning for Complex Data  |  Biometrics  |  Statistical methods for visual recognition

Anderson Rocha
About my co-author

• Assistant Professor at the University of Notre Dame
  - Ph.D. from the University of Colorado 2009
  - 2012 — 2015 Harvard University Center for Brain Science

• Research in Computer Vision and Machine Learning

Reverse engineering biological vision

Tools for Neuroscience

Statistical methods for visual recognition

Digital Humanities

Walter J. Scheirer
Organization

- 3-hour, 4-part tutorial
  - **Part 1:** An introduction to the open set recognition problem
  - **Part 2:** Algorithms that minimize the risk of the unknown
  - **Part 3:** Case studies – visual computing and other areas
  - **Part 4:** Research opportunities and trends
- Ask questions on-the-fly
- 15-minute break after the first half part
Available online

• This tutorial and some source-code mentioned herein are available online at:

  http://recodbr.wordpress.com/

• In case you have any questions, **send us an e-mail**
Part 1: An Introduction to the Open Set Recognition Problem
Benchmarks in Machine Learning

Assume we have examples from all classes:

- Caltech 256
  - airplanes
  - elephant
  - soccer ball
  - car

- water lily
Out in the real world…

Be on the lookout for blue Ford sedans

while rejecting the trees, signs, telephone poles…

What is the general recognition problem?

- Duin and Pekalska*: how one should approach multi-class recognition is still an open issue
  - Is it a series of binary classifications?
  - Is it a search performed for each possible class?
  - What happens when some classes are ill-sampled, not sampled at all or undefined?

Open Space in Classification
“There are known knowns…”

- **known classes**: the classes with distinctly labeled positive training examples (also serving as negative examples for other known classes)

- **known unknown classes**: labeled negative examples, not necessarily grouped into meaningful categories

- **unknown unknown classes**: classes unseen in training
Definitions

• **Closed Set Recognition:** all testing classes are known at training time

• **Open Set Recognition:** incomplete knowledge of the world is present at training time, and unknown classes can be submitted to an algorithm during testing
The burden for information forensics and security

• Closed set results look better than they really are, which misleads practitioners

• “Off-the-shelf” classifiers are not sufficient to solve the open set problem

• Open set problems are found in nearly every case where recognition algorithms are present
Fingerprint Spoof Detection

LivDet 2011 - Closed Set vs. Open Set Evaluation

<table>
<thead>
<tr>
<th>Training materials</th>
<th>$\mathcal{L}_{BSIR}$</th>
<th>EER$_{known}$ [%]</th>
<th>EER$_{novel}$ [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Latex+EcoFlex</td>
<td>3.9</td>
<td>13.8</td>
<td></td>
</tr>
<tr>
<td>WoodGlue+Latex</td>
<td>11.0</td>
<td>12.2</td>
<td></td>
</tr>
<tr>
<td>Gelatine+Latex</td>
<td>8.8</td>
<td>14.9</td>
<td></td>
</tr>
<tr>
<td>Silgum+Latex</td>
<td>6.0</td>
<td>16.2</td>
<td></td>
</tr>
<tr>
<td>EcoFlex+Silgum</td>
<td>10.8</td>
<td>23.6</td>
<td></td>
</tr>
<tr>
<td>Gelatine+EcoFlex</td>
<td>13.9</td>
<td>20.9</td>
<td></td>
</tr>
<tr>
<td>Silgum+Gelatine</td>
<td>13.2</td>
<td>20.7</td>
<td></td>
</tr>
<tr>
<td>WoodGlue+Silgum</td>
<td>14.7</td>
<td>19.3</td>
<td></td>
</tr>
<tr>
<td>Gelatine+WoodGlue</td>
<td>12.4</td>
<td>15.4</td>
<td></td>
</tr>
<tr>
<td><strong>Average EER:</strong></td>
<td><strong>10.5</strong></td>
<td><strong>17.5</strong></td>
<td></td>
</tr>
</tbody>
</table>

Rattani, Scheirer and Ross, T-IFS 2015

Yambay et al. LivDet 2011
Open Set Fingerprint Spoof Detection

LivDet 2011 - competitors had error rates approximately 3.5x larger for unknown recipes
Steganalysis

- Pevny et al. T-IFS 2010
- Steganalysis in the spatial domain is very sensitive to the types of cover images
- Cross data set evaluation
  - 3 known classes, 1 unknown class

Steganalysis

Open Set

<table>
<thead>
<tr>
<th></th>
<th>bpp</th>
<th>CAMERA</th>
<th>BOWS2</th>
<th>JPEG85</th>
<th>NRCS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disjoint</td>
<td>0.25</td>
<td>0.3388</td>
<td>0.1713</td>
<td>0.3247</td>
<td>0.3913</td>
</tr>
<tr>
<td>Disjoint</td>
<td>0.5</td>
<td>0.2758</td>
<td>0.1189</td>
<td>0.2854</td>
<td>0.3207</td>
</tr>
<tr>
<td>Joint</td>
<td>0.25</td>
<td>0.0910</td>
<td>0.0845</td>
<td>0.0198</td>
<td>0.2013</td>
</tr>
<tr>
<td>Joint</td>
<td>0.5</td>
<td>0.0501</td>
<td>0.0467</td>
<td>0.0102</td>
<td>0.08213</td>
</tr>
</tbody>
</table>

Closed Set
A surprising finding…
Read-out layer

Typical CNN architecture

Softmax

\[ P(y = j | x) = \frac{e^{v_j(x)}}{\sum_{i=1}^{N} e^{v_i(x)}} \]

Sum over all of the classes

Linear SVM

\[ \min \frac{1}{2} ||w||^2 \]

subject to

\[ y_i(w \cdot x_i + b) \geq 1, \forall_i \]

Known positive or negative sample

Cosine Similarity

\[ \frac{A \cdot B}{||A|| \, ||B||} < \delta \]

Threshold determined empirically via known pairs

\[ \mathbb{A} \mathbb{B} \]
Evolving images to match CNN classes

But you don’t have to use tricky manipulations

GoogleNet Output

Label: Hammerhead Shark

Label: Blow Dryer

Label: Mosque

Label: Syringe

Label: Trimaran

Label: Missile
Are performance measures misleading us?
Psychophysics on the Model

Psychophysics pipeline

1. Render Class
   Canonical View (CCV)
   Candidates

2. CCV Classifier

3. Manipulate Chosen
   Variable

4. Classify Images

5. Generate
   Psychometric Curve

Accuracy  vs  Area Visible

Fish
Plane
Skyscraper
What standard options do we have to solve this problem?
Binary Classification
Multi-class 1-vs-All Classification
1-class Classification

“All positive examples are alike; each negative example is negative in its own way”

Zhao and Huang (with some help from Tolstoy)
CVPR 2001
Vision problems in order of “openness”

- Multi-class Classification: Training and testing samples come from known classes.
- Face Verification: Claimed identity, possibility for impostors.
- Detection: One class, everything else in the world is negative.
- Open Set Recognition: Multiple known classes, many unknown classes.

Let’s formalize openness

$$\text{openness} = 1 - \sqrt{\frac{2 \times |\text{training classes}|}{|\text{testing classes}| + |\text{target classes}|}}$$
## Examples of openness values

<table>
<thead>
<tr>
<th></th>
<th>Targets</th>
<th>Training</th>
<th>Testing</th>
<th>Openness</th>
</tr>
</thead>
<tbody>
<tr>
<td>Typical Multi-class</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>0%</td>
</tr>
<tr>
<td>Face Verification</td>
<td>12</td>
<td>12</td>
<td>50</td>
<td>38%</td>
</tr>
<tr>
<td>Typical Detection</td>
<td>1</td>
<td>100,000</td>
<td>1,000,000</td>
<td>55%</td>
</tr>
<tr>
<td>Object Recognition</td>
<td>88</td>
<td>12</td>
<td>88</td>
<td>63%</td>
</tr>
<tr>
<td>Object Recognition</td>
<td>88</td>
<td>6</td>
<td>88</td>
<td>74%</td>
</tr>
<tr>
<td>Object Recognition</td>
<td>212</td>
<td>6</td>
<td>212</td>
<td>83%</td>
</tr>
</tbody>
</table>
Fundamental multi-class recognition problem

\[
\text{argmin}_f \left\{ R_{\mathcal{I}}(f) := \int_{\mathbb{R}^d \times \mathbb{N}} L(x, y, f(x)) P(x, y) \right\}
\]

Ideal Risk  Loss Function  Joint Distribution

Undefined for open set recognition!

Open Space

Positives

Negatives
Open Space

• Open space is the space far from known data

• We need to address the infinite half-space problem of linear classifiers

• Principle of Indifference*
  - If there is no known reason to assign probability, alternatives should be given equal probability
  - One problem: we need the distribution to integrate to 1!

Open Space Risk

Open Space Risk: the relative measure of open space to the full space

$$R_O(f) = \frac{\int_{O} f(x) dx}{\int_{S_O} f(x) dx}$$

open space + positive training examples
The open set recognition problem

Preliminaries
Space of positive class data: \( P \)
Space of other known class data: \( K \)
Positive training data: \( \hat{V} = \{v_1, \ldots, v_m\} \) from \( P \)
Negative training data: \( \hat{K} = \{k_1, \ldots, k_n\} \) from \( K \)
Unknown negatives appearing in testing: \( U \)
Testing data: \( \hat{T} = \{t_1, \ldots, t_z\} \), \( t_i \in P \cup K \cup U \)

Assume the problem openness is \( > 0 \)
The open set recognition problem

Minimize open set risk:

\[ \arg \min_{f \in \mathcal{H}} \left\{ R_O(f) + \lambda_r R_E(f(\hat{V} \cup \hat{K})) \right\} \]

- Open Space Risk Associated with \( \mathcal{U} \)
- Regularization Constant
- Empirical Risk Function
What’s missing from our definition of open space risk?

\[ R_\mathcal{O}(f) = \frac{\int_{\mathcal{O}} f(x) \, dx}{\int_{\mathcal{S}_\mathcal{O}} f(x) \, dx} \]

open space

Open space + positive training examples

The definition doesn’t tell us how to define \( \mathcal{O} \)
Incorporating open space risk into a model

- Discriminative models?
  
  Don’t address unknown unknowns in open space

- Generative models?
  
  Don’t address unknown unknowns in open space

- Hard negative mining (Felzenszwalb et al. 2010)?
  
  Not possible to mine examples from unknown classes
Abating Process

- Model enforced decay of probability away from supporting evidence

Monotonically decreasing prob.

Positive training data

The Compact Abating Probability Model

**Conceptual example:** if we are labeling location data using training data only from Campinas, Brazil, it would be risky to apply that model to South Bend, Indiana.

Idea: ensure that the recognition function is decreasing away from the training data, so that thresholding it limits the labeled region.
Definition of Open Space

\[ \mathcal{O} = S_0 - \bigcup_{i \in N} B_r(x_i) \]

closed ball of radius \( r \)
centered around training sample \( x_i \)

Treat \( r \) as a problem-specific parameter
Abating Bound

\[ \forall x, x_i : \quad 0 < K(x, x_i) \leq A(\|x - x_i\|) \]

- Positive Definite Kernel (e.g., RBF)
- Non-negative finite square integrable continuous decreasing function

When \( \forall x, \exists x^* \mid f(x) \leq A(\|x - x^*\|) \),
\( f \) is abating because the spatial influence decreases with distance from \( x^* \)
Abating Probabilistic Point Model

$$M(x) = p_f(F(K(x, x_1) \ldots K(x, x_m)); y)$$

Fusion Operator (e.g., sum or product)

Model

Probability of points associating becomes less intense as the spatial separation of any two points increases.
Fused Abating Property

After fusion there is an abating bound function centered at $x_0$ such that the fused value $F$ is bounded from above by that abating function.

$$F(K(x, x_1) \ldots K(x, x_m)) \leq A_{x'}(\|x' - x\|)$$

Abating Bound Function
Compact Abating Probability (CAP) Model

\[
\min_{x_i \in \mathcal{K}} \| x - x_i \| > \tau \Rightarrow M_{\tau}(x) = 0
\]

Features beyond a given thresholded \( \tau \) from the closest training point have zero probability.
Goal: Multi-class Open Set Recognition
Model: Compact Abating Probability

- Monotonically decreasing prob.
- Threshold on prob.
- Prob. from kernel machine varies locally with distance to training points

Class ‘3’

- $P(3|3) > \delta_K$
- $P(3|?) < \delta_K$

CAP thresholded region
EVT-101: Statistical Extreme Value Theory for Visual Recognition
The Statistical Extreme Value Theory (EVT)

Why EVT for visual recognition problems?

• Powerful explanatory theory (Scheirer et al. T-PAMI 2011)

• Effective tool for statistical modeling of decision boundaries (Broadwater et al. IEEE T. Signal Processing 2010, Fragoso and Turk CVPR 2013)
  - Calibration models (Scheirer et al. ECCV 2010)
The Extreme Value Theorem

Let \((s_1, s_2, \ldots, s_n)\) be a sequence of i.i.d. samples. Let 
\(M_n = \max \{s_1, \ldots, s_n\}\). If a sequence of pairs of real 
numbers \((a_n, b_n)\) exists such that each \(a_n > 0\) and

\[
\lim_{x \to \infty} P \left( \frac{M_n - b_n}{a_n} \leq x \right) = F(x)
\]

then if \(F\) is a non-degenerate distribution function, it 
belongs to one of three extreme value distributions\(^1\).

The i.i.d. constraint can be relaxed to a weaker 
assumption of exchangeable random variables\(^2\).

---

2. S. Berman, “Limiting Distribution of the Maximum Term in Sequences of Dependent Random Variables,” 
The Weibull Distribution

The sampling of the top-\(n\) scores always results in an EVT distribution, and is **Weibull** if the data are bounded\(^1\).

\[
f(x; \lambda, k) = \begin{cases} 
 \frac{k}{\lambda} \left( \frac{x}{\lambda} \right)^{k-1} e^{-\left(\frac{x}{\lambda}\right)^k} & x \geq 0 \\
0 & x < 0 
\end{cases}
\]

Choice of this distribution is not dependent on the model that best fits the entire non-match distribution.

Fitting an EVT Distribution

• EVT applies regardless of the overall distribution
• Sampling the extrema in the tail of an overall distribution always results in an EVT distribution
Is there a difference between central tendency modeling and EVT?

- Sample set of 1,000 values from a standard normal distribution
- Compute means over 10,000 trials

What does the histogram look like?
Bell curve
What if we’re interested in extrema points instead?

- Sample set of 1,000 values from a standard normal distribution
- Retain the *maximums* over 10,000 trials

What does the histogram look like?
The peak is now at 3.2, and there is noticeable skew
Probability estimation (R): central tendency

Means from the 10,000 trials

```
library("MASS")
fitdistr(bufferMean, "normal")
#   mean      sd
# -0.0001344194 0.0313828480
# (0.0003138285) (0.0002219102)
```

Improbable outcome

```
pnorm(0, -0.0001344194, 0.0313828480, FALSE)
# [1] 0.4982913

pnorm(1, -0.0001344194, 0.0313828480, FALSE)
# [1] 3.611041e-223

pnorm(2, -0.0001344194, 0.0313828480, FALSE)
# [1] 0
```
Probability estimation (R): EVT

Maxima from the 10,000 trials

```
library(SpatialExtremes)
gevmle(bufferMax)

<table>
<thead>
<tr>
<th>loc</th>
<th>scale</th>
<th>shape</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.08305916</td>
<td>0.29802546</td>
<td>-0.07158273</td>
</tr>
</tbody>
</table>
```

Probable outcome

```
pgev(0, 3.08305916, 0.29802546, -0.07158273, FALSE) #[1] 1
pgev(1, 3.08305916, 0.29802546, -0.07158273, FALSE) #[1] 1
pgev(2, 3.08305916, 0.29802546, -0.07158273, FALSE) #[1] 1
pgev(3, 3.08305916, 0.29802546, -0.07158273, FALSE) #[1] 0.7322732
pgev(4, 3.08305916, 0.29802546, -0.07158273, FALSE) #[1] 0.03047941
```
A good alternative to central tendency modeling

(a) Binary Discriminative Model

(b) Per class Gaussian Model + Bayesian decision

(c) EVT Fit for the min and max tail of each class + Bayesian decision
Example two-category discrimination task along a parametric stimulus axis.
Extrema as visual features

Tanaka et al. - Atypicality

Leopold et al. - Caricaturization

Tanaka and Farah - Visual Attributes

Itti et al. - Visual Saliency

Barenholtz and Tarr - Part Boundaries


How does EVT apply to computer vision?
Meta-Recognition Theory

Meta-recognition is **recognizing when a recognition system is working or failing**. It is important for threshold selection, failure prediction and improving fusion.

Failure Prediction

Can we recognize, in some automated fashion, if a recognition system result is a success or a failure?

If so, can we quantify the probability of success or failure?
Meta-Recognition as failure prediction
Predicting Failures of Vision Systems

Zhang et al. CVPR 2014

Learn conditions that cause a target algorithm to fail

Statistical EVT Failure Prediction

• Get scores, sort and take top $N$

• Fit an extreme value distribution to get model of non-match distribution, exclude top score

• Determine if top score is outlier from distribution, If so predict success. Else predict failure

• Detect outlier using fraction CDF below the potential outlier. 99.999999% is a good test!
Statistical EVT Failure Prediction

Overall Distribution of Scores

Portfolios of Gallery

Match

Distribution’s tail

Extrema

Extreme Value

Tail Analysis

Best of Portfolio Matches

Portfolios
EVT-based failure prediction

• Using meta-recognition, evaluated 12 algorithms across 4 problems. Always significantly better than simple thresholds on score.

  ✓ Face Biometrics
  ✓ Fingerprint Biometrics
  ✓ Multi-biometric fusion
  ✓ SIFT + earth-mover distance based object recognition
  ✓ Content-based image retrieval (4 algorithms)

• Led to new fusion algorithm, better than traditional algorithms on all datasets considered.
Examined impact of i.i.d. assumptions and sizes of top-N data needed for prediction
What else can Meta-Recognition do?

Decision fusion (fuse only those that are not predicted to fail) or weighted score fusion.

For statistical EVT prediction use:

w-score fusion where:

\[ w\text{-score}(x) = \text{CDFWeibull}(x) \]

Use w-score to weight data for fusion, i.e., compute average w-score over different algorithms/modalities.

w-score normalization

**Require:** a collection of scores $S$, of vector length $m$, from a single recognition algorithm $j$;

1. **Sort** and retain the $n$ largest scores, $s_1, \ldots, s_n \in S$;

2. **Fit** a Weibull distribution $W_S$ to $s_2, \ldots, s_n$, skipping the hypothesized outlier;

3. **While** $k < m$ do

4. $s'_k = \text{CDF}(s_k, W_S)$

5. $k = k + 1$

6. end while
Fusion Performance

w-score fusion outperformed z-score with sum (or product) fusion on all experiments. In general, the lower the performance the greater the differential gain.
Fusion Problems For:

• Existing theories for fusion algorithms presume consistent data and work to address noise. What happens when user intentionally attempts to thwart the system by changing/destroying their data?

• What is needed is an approach to predict when a particular modality/algorithm is failing and then ignore it.
Failure Prediction and Fusion

Traditional Fusion can be degrade system performance, especially when adversaries try to defeat it.

Meta-recognition can predict and select correct modality.

Ramirez Abadia

Meta-recognition is a useful mathematical theory for fusion that predicts “failing” data
Classic fusion can make things worse!

BSSR1 has only 600 paired sets of data and was too easy. So we made chimera data, mixing all fingers and faces (6000 samples).

This shows the real power of MR– automatically ignoring bad data!
## Failure-prediction W-score fusion vs z-score

<table>
<thead>
<tr>
<th>W</th>
<th>Z</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>81.6</td>
<td>65.2</td>
<td>face C (impostor), finger L1</td>
</tr>
<tr>
<td>88.1</td>
<td>67.4</td>
<td>face C (impostor), finger R1</td>
</tr>
<tr>
<td>81.6</td>
<td>65.9</td>
<td>face G (impostor), finger L1</td>
</tr>
<tr>
<td>88.1</td>
<td>68.1</td>
<td>face G (impostor), finger R1</td>
</tr>
<tr>
<td>73.3</td>
<td>58.0</td>
<td>face C (impostor), face G (impostor), finger L1</td>
</tr>
<tr>
<td>79.8</td>
<td>60.6</td>
<td>face C (impostor), face G (impostor), finger R1</td>
</tr>
</tbody>
</table>

3000 samples from NIST BSSR1 data

**Rank 1 fusion with z-scores is highly impacted by failing modalities; the failure prediction fusion with w-scores is very close to rank 1 of the modality that isn’t failing, even with multiple failures.**
• Probability calibration is only well defined close to the decision boundary (Bartlett and Tewari JMLR 2007)

• Boundary is defined by the training samples that are effectively extremes,
  - Calibration models should be based on EVT
Calibration for decision boundaries

1. Get tail of decision scores from the opposite class
2. Fit Weibull to values in the tail:
   \[ f(x; \lambda, k) = \begin{cases} \frac{k}{\lambda} x^{k-1} e^{-(x/\lambda)^k} & x \geq 0 \\ 0 & x < 0 \end{cases} \]
3. Compute normalized scores using CDF of the Weibull:
   \[ F(x; k, \lambda) = 1 - e^{-(x/\lambda)^k} \]

Fusion after normalization

1. maximize over \( I \)
2. subject to \( A_j(I) = F(T(s_j(I)); W_j) \)
3. for \( \forall j \in J \) satisfying \( 0 \leq a_j \leq A_j(I) \leq \beta_j \leq 1 \)

The goal is to find images \( I \) that maximize the \( L_1 \) norm for each attribute \( j \) in the query set \( J \).

Multi-Attribute Search

Target Attribute Similarity Search

“Indian Females”

“Male and Black Hair Like Target”

“Male”

“Indian”

“Target”
Utility of the calibration model
Part 2: Algorithms that Minimize the Risk of the Unknown
Kernel Density Estimation (KDE)


1. Fit a Gaussian distribution to the positive training data for a class

2. Empirically estimate a threshold $\tau$ over the resulting density
Kernel Density Estimation

Comparison of 1D bandwidth selectors

\( \tau_0 \)

\( \tau_1 \)

Density function

Comparison of 1D bandwidth selectors \( \odot \) BY-SA 3.0 Drleft
KDE Pitfalls

• Nearly always results in overfitting for visual recognition problems

• Choice of Gaussian distribution questionable in many circumstances

• How do we estimate a good $\tau$?
Support Vector Data Description (SVDD)


- Hypersphere with the minimum radius is estimated around the positive class data that encompasses almost all training points.
Support Vector Data Description (SVDD)

Image credit: Shen et al. Sensors 2012
Support Vector Data Description (SVDD)

- Sensitive to feature scaling (Tax and Duin ASCI 2002)
- Difficult to solve using good numerical optimization (Chang et al. NTU Tech. Report 2013)
- Far less effective than binary classifiers when some sampling of negatives is available
  - Overfits the training data
1-Class SVM

- Only positive data at training time
- “Origin” defined by the kernel serves as the only member of a “second class”
- Training object yields a binary classifier \( f \)
- When used, usually for outlier or anomaly detection
1-Class SVM

Image credit: L. Manevitz and M. Yousef, “One-Class SVMs for Document Classification” JMLR 2001
1-Class SVM Objective

\[
\begin{aligned}
\min & \frac{1}{2} \| w \|^2 + \frac{1}{\nu m} \sum_{i=1}^{l} \xi_i - \rho \\
\text{subject to} & \quad (w \cdot \Psi(x_i)) \geq \rho - \xi_i \quad i = 1, 2, \ldots, m \quad \xi_i \geq 0
\end{aligned}
\]

\(\nu\) controls the upper bound on training error
1-Class SVM Implementation

LIBSVM (linear and RBF)


options:
-s svm_type : set type of SVM (default 0)
  0 -- C-SVC     (multi-class classification)
  1 -- nu-SVC    (multi-class classification)
  2 -- one-class SVM
  3 -- epsilon-SVR (regression)
  4 -- nu-SVR    (regression)
Why didn’t the 1-class SVM catch on?

• Zhou and Huang *Multimedia Systems* 2003
  - Kernel and parameter selection
    ‣ Gaussian kernels lead to over-fitting
    ‣ Parameters chosen in *ad hoc* fashion
    ‣ An issue in other domains too!
Problems with Existing Models for Binary Problems

Binary SVs

Binary RBF SVM Decision Boundary

1-class SVs

1-class RBF SVM Decision Boundary

1-class RBF SVM
Other machine learning approaches


Open World Recognition

- World with Knowns (K) & Unknowns Unknowns (UU)
- Label Data
  - NU: Novel Unknowns
  - LU: Labeled Unknowns
- Incremental Class Learning
- K: Known
- Scale

Out in the Real-World

- Chess board
- Elephant
- Soccer ball
- Car
- Pool Table
- Bowling Pin
- Calculator
- Detect New Category
- Water lily
- Boxing glove
- Airplanes
- Calculator
- 256

Questions:
- ?
- ?
- ?
Related Work

Scalable Learning

Liu+ (CVPR’13)

Deng+ (NIPS’11)

Marszalek+ (ECCV’08)

Scheirer+ PAMI’13

Jain+ (ECCV’14)

Open World Recognition

Yeh+ (CVPR’08)

Li+ CVPR’07

Scheirer+ PAMI’14

Incremental Learning

Ristin+ (CVPR’14)

Mensink+ (PAMI’13)

Open Set Learning

Open Set Learning

103
Slab Model

Negatives

A

Positives

Ω
Base Linear 1-vs-Set Machine
Generalization
Specialization
Open space risk for linear slab model

\[ \delta_A \quad \text{Marginal distance of near plane} \]

\[ \delta_{\Omega} \quad \text{Marginal distance of far plane} \]

\[ \delta^+ \quad \text{Separation needed to account for all positive data} \]

\[ \frac{\delta_{\Omega} - \delta_A}{\delta^+} \quad \text{Overgeneralization risk} \]

\[ \frac{\delta^+}{\delta_{\Omega} - \delta_A} \quad \text{Overspecialization risk} \]
Open space risk for linear slab model

\[ R_\varsigma = \frac{\delta_\Omega - \delta_A}{\delta^+} + \frac{\delta^+}{\delta_\Omega - \delta_A} + p_A \omega_A + p_\Omega \omega_\Omega \]

Two additional terms

- Importance of open space around \( A \)
- Importance of open space around \( \Omega \)
- Margin around \( A \)
- Margin around \( \Omega \)
Training and testing data

Space of positive class data: $\mathcal{P}$
Space of other known class data: $\mathcal{K}$
Positive training data: $\hat{\mathcal{V}} = \{v_1, \ldots, v_m\}$ from $\mathcal{P}$
Negative training data: $\hat{\mathcal{K}} = \{k_1, \ldots, k_n\}$ from $\mathcal{K}$
Unknown negatives appearing in testing: $\mathcal{U}$
Testing data: $\mathcal{T} = \{t_1, \ldots, t_z\}, t_i \in \mathcal{P} \cup \mathcal{K} \cup \mathcal{U}$
Sketch of the 1-vs-Set Machine training algorithm

1. Train a linear SVM $f$ using $\hat{V}$ and $\hat{K}$

2. Generate decision scores for each training point in $\hat{V}$ and $\hat{K}$

3. Sort decision scores, where $s_k$ is the minimum and $s_j$ is the maximum

4. Initialize $A$ to margin plane of $f$, and $\Omega$ to $s_j$

5. Iteratively move $A$ to $s_{k+1}$ or $s_{k-1}$, $\Omega$ to $s_{j-1}$ or $s_{j+1}$ to minimize $R_\xi(f) + \lambda_r R_\varepsilon$
1-vs-Set Machine Plane Refinement

Positive Pressure \( p_A > 0 \)

Plane \( A \) after initial optimization

Negative Pressure \( p_A < 0 \)

Plane \( A \) after refinement with \( p_A = -0.5 \)

\( \omega_A \)
function PREDICT(t_x, f, A, Ω) 
    if (A ≤ f(t_x) and f(t_x) ≤ Ω) then Return +1 
    else Return -1 
end if 
end function
What could be better about the the 1-Vs-Set Machine?

• Does not inherently support multi-class open set recognition
• Does not support non-linear kernels
• Does not contain a CAP model
• Lack of calibrated probability scores
The Statistical Extreme Value Theory (EVT)

Why EVT for visual recognition problems?

• Powerful explanatory theory (Scheirer et al. T-PAMI 2011)

• Effective tool for statistical modeling of decision boundaries (Broadwater et al. IEEE T. Signal Processing 2010, Fragoso and Turk CVPR 2013)
  - Calibration models (Scheirer et al. ECCV 2010)
$P_I$-SVM: Modeling Probability of Inclusion

- Fit a robust single-class probability model over the positive class scores from a discriminative binary classifier
  - Binary (RBF) classifier helps discriminate the positive class from the known negative classes
  - Single-class probability model adjusts decision boundary to avoid misclassification of “unknowns”
Consider a kernelized SVM

\[ h(x) = \sum_{i=1}^{n} y_i \alpha_i K(x_i, x) + b \]
Fit model to tail of positive side of decision boundary

Non-Match Data (Negative Side)

Weibull Fit to Match Data

parameters = $\theta_y = \tau_y, \kappa_y, \lambda_y$
Probability model for inclusion

$$P_I(y|x, \theta_y) = \xi \rho(y) P_I(x|y, \theta_y) = \xi \rho(y) \left(1 - e^{-\left(\frac{x - \tau_y}{\lambda_y}\right)^{\kappa_y}}\right)$$

- Prior prob. of class $y$
- Constant
- Weibull CDF defined by $\theta_y$
Unnormalized Posterior Estimate

If all classes and priors are known, then Bayes’ theorem yields:

$$\xi = \frac{1}{\sum_{y \in C} \rho(y) P_I(x|y, \theta_y)}$$

But this isn’t true for open set recognition, so we let $\xi = 1$ and treat the posterior estimate as unnormalized.
Multi-class Open Set Recognition with $P_I$-SVM

$$y^* = \arg\max_{y \in \mathcal{C}} P_I(y|x, \theta_y) \quad \text{subject to} \quad P_I(y^*|x, \theta_{y^*}) \geq \delta$$

Min. threshold on class probability
Tail Size Estimation

• EVT tells us how to model extrema, but says nothing about how many samples to model
  - The difference between a tail size of 5% and a tail size of 20% can produce a difference in recognition accuracy of 15-20%
  - Need **automatic estimation**
Support Vectors as Extrema

• Support vectors are a type of extreme sampling that effectively describes the class boundary

• Is there a known parametric relationship between training data size, dimensionality, and the number of support vectors? No

Alternative: consider extrema to be the points close to the original decision boundary and count them
Tail size estimation

Indicator Function

\[ B^+(x; \epsilon) = \begin{cases} 
1 & \text{if } h(x) \leq \epsilon \\
0 & \text{otherwise} 
\end{cases} \]

When \( \epsilon > 0 \), some points inside the positive boundary included

Tail size approximation:

\[ \hat{T}^+_\epsilon = \max(3, \psi \times |\alpha^+|) \]

\[ \psi \in [1.25 - 2.5] \]

Positive Tail Size

\[ T^+_\epsilon = \sum_{x \in \mathcal{M}_y} B^+(x; \epsilon) \]
Tail size estimation

First class folds

All class folds
Normalized decision scores for $P_l$-SVM
$P_I$-SVM Implementation

Patch to LIBSVM available at:
https://github.com/ljain2/libsvm-openset


options:
-s svm_type : set type of SVM (default 0)
  0 -- C-SVC
  1 -- nu-SVC
  2 -- one-class SVM
  3 -- epsilon-SVR
  4 -- nu-SVR
  5 -- open-set oneclass SVM (open_set_training_file required)
  6 -- open-set pair-wise SVM (open_set_training_file required)
  7 -- open-set binary SVM (open_set_training_file required)
  8 -- one-vs-rest WSVM (open_set_training_file required)
  9 -- One-class PI-OSVM (open_set_training_file required)
 10 -- one-vs-all PI-SVM (open_set_training_file required)
Is PI-SVM what we’re looking for for open set recognition?

• Pros:
  + Supports multi-class open set recognition
  + Better generalization than the 1-vs-Set Machine

• Cons:
  - One-sided calibration model (just probability of inclusion)
  - Does not make use of a CAP model
Let $d_x$ be the distance to the nearest neighbor of $x$

Let $d_x > \tau \Rightarrow p_a(x) = 0$ and $p_a(x) = \frac{\left|\tau - d_x\right|}{\tau}$

In a multi-class setting, this results in a thresholded NN algorithm that can reject an input as unknown.
NN+CAP

• Pros:
  + With sufficiently dense sampling, NN+CAP reduces to NN
  + Limiting error of no more than twice the Bayes error rate
  + Simple to train

• Cons:
  - Weak probability model
The Weibull-calibrated SVM (W-SVM)

• Binary SVMs are better than 1-Class SVMs - how do they fit into the context of CAP models?

• Unfortunately, the decision score isn’t a canonical sum. But calibration is possible (Hoffman et al. Annals of Stat. 2008):

  1. Collect all positive coefficients in one sum
  2. Collect all negative coefficients into another sum
  3. Split the bias between them
  4. View SVM as applying a decision rule over which is more similar

Binary RBF SVM incorporating a CAP model

• Combine probabilities computed for both 1-class and binary RBF SVMs

• 1-class SVM CAP model is a conditioner

If \( P_O(y|x) > \delta_t \), then consider \( P_O(y|x) \)
else reject

could be very small
Dual tail fitting

Separating positive and negative data is useful

Assume a set of known classes $Y$

For a class $y \in Y$, we can use positive scores from $y$ to estimate $P^+(y|x)$.

We can use negative scores from other known classes to estimate $P(y \setminus y | x)$. 
Dual tail fitting

Reverse Weibull Fit to Non-Match Data

Weibull Fit to Match Data

Density of non-match scores

Density of match scores

SVM Decision Boundary

Known Negative Classes = '0', '1', '2'

\[ \lambda, \eta, \nu, \psi, \kappa \]

\[ \delta R = 0.5 \times \text{openness} \]
Dual tail fitting

Closed set scenario: $P^+(y|x) = 1 - P^-(y \backslash y | x)$

In an open set scenario, we can’t make the above assumption.

To minimize open set risk, $P^+$ and $P^-$ are considered only when $P_O(y|x) > \delta_t$
EVT Parameters

• Reverse Weibull and Weibull are defined by three parameters
  - location $\nu$, scale $\lambda$, and shape $\kappa$

• Maximum Likelihood Estimation to estimate the best fits for $\eta$ and $\psi$
  - $\nu_\eta$, $\lambda_\eta$, $\kappa_\eta$
  - $\nu_\psi$, $\lambda_\psi$, $\kappa_\psi$
Two independent estimates for $P(y \mid f(x))$

Weibull CDF from match data

$P_\eta(y \mid f(x)) = 1 - e^{-\left(\frac{f(x) - \nu_\eta}{\lambda_\eta}\right)^{\kappa_\eta}}$

Reverse Weibull CDF from non-match data

$P_\psi(y \mid f(x)) = e^{-\left(\frac{f(x) - \nu_\psi}{\lambda_\psi}\right)^{\kappa_\psi}}$
Combining probability estimates

Two options:

\( P_\eta \times P_\psi \): the probability that the input is from the positive class AND NOT from any of the known negative classes.

\( P_\eta + P_\psi \): either a positive OR NOT a known negative.

For open set recognition, \( P_\psi \) should be modulated by other supporting evidence of the sample being positive. Product is the preferred combo.
Multi-class W-SVM recognition

Indicator variable: \[ \iota_y = 1 \text{ if } P_O(y|x) > \delta_{\tau} \]

\[ y^* = \arg\max_{y \in \mathcal{Y}} P_{\eta,y}(x) \times P_{\psi,y}(x) \times \iota_y \]

subject to \[ P_{\eta,y^*}(x) \times P_{\psi,y^*}(x) \geq \delta_R \]
Training a W-SVM Step-by-Step

- For simplicity, let’s focus on a single class ("3")
- Two SVM models (1-class and binary)
- Three EVT distribution fits
- The collection of SVM models, EVT distribution parameters, and thresholds constitute the W-SVM.
Step 1: Train a 1-class SVM $f^o$

Class Label = ‘3’

RBF one-class SVM yields a CAP model
Step 2: Fit Weibull over tail of scores from $f^o$

Weibull Fit to Match Data

Class ‘3’

Density of match scores

$\lambda_o, \nu_o, \kappa_o$

$\delta_T = 0.001$
Step 3: Train a binary SVM $f$

Class Label = ‘3’

Known Negative Classes = ‘0’, ‘1’, ‘2’
Step 4: Fit EVT distributions over tails of scores from $f$

Density of non-match scores

Reverse Weibull Fit to Non-Match Data

Weibull Fit to Match Data

Density of match scores

Known Negative Classes = ‘0’, ‘1’, ‘2’

Class ‘3’

$\delta_R = 0.5 \times openness$

$\lambda_{\psi}, \nu_{\psi}, \kappa_{\psi}$

$\lambda_{\eta}, \nu_{\eta}, \kappa_{\eta}$
W-SVM testing (known class)

• Let’s focus on the class we just trained for (“3”)
• Six steps are necessary to test the input
• Assume four known classes (“0”, “1”, “2”, “3”)
Step 1: Apply 1-class SVM CAP model for all known classes

Input: \( x = 3 \)

\[
\begin{align*}
f_0^o(x) &= s_0 & f_1^o(x) &= s_1 \\
f_2^o(x) &= s_2 & f_3^o(x) &= s_3
\end{align*}
\]
Step 2: Normalize all 1-class SVM scores using EVT models

$$\lambda_{o,0}, \nu_{o,0}, \kappa_{o,0} \quad \lambda_{o,1}, \nu_{o,1}, \kappa_{o,1}$$

Apply CDF for each class to each score

Probability model for test instance: $P_o$
Step 3: Test probabilities

\[ P_0(0|x) < \delta_\tau, \; l_0 = 0; \quad P_0(1|x) < \delta_\tau, \; l_1 = 0; \]
\[ P_0(2|x) > \delta_\tau, \; l_2 = 1; \quad P_0(3|x) > \delta_\tau, \; l_3 = 1 \]
Step 4: Apply binary SVMs

\[ f_2(x) = s_2 \quad \quad f_3(x) = s_3 \]
Step 5: Normalize all binary SVM scores using EVT match and non-match models

Apply 2 CDFs per class for each score
Step 6: Fuse and test probabilities

\[
P_{\eta,0}(x) \times P_{\psi,0}(x) \times \iota_0 = 0 < \delta_R
\]

\[
P_{\eta,1}(x) \times P_{\psi,1}(x) \times \iota_1 = 0 < \delta_R
\]

\[
P_{\eta,2}(x) \times P_{\psi,2}(x) \times \iota_2 = 0.001 < \delta_R
\]

\[
P_{\eta,3}(x) \times P_{\psi,3}(x) \times \iota_3 = 0.877 > \delta_R
\]
Models for class ‘3’ and the data point for this example

Monotonically decreasing prob. bound

Threshold on prob.

Prob. from kernel machine varies locally with distance to training points

P(3|\mathcal{D}) > \delta_R

CAP thresholded region

W-SVM thresholded region
W-SVM testing (unknown class)

- Assume four known classes ("0", "1", "2", "3")
- Consider as input a member of a class that is different from the training data ("Q")
  - This point will fall outside of the CAP thresholded region (i.e., it exists in open space)
- Four steps are necessary to reject the input
Step 1. Apply 1-class SVM CAP model for all known classes

Input: $x = Q$

$f^o_0(x) = s_0 \quad f^o_1(x) = s_1$

$f^o_2(x) = s_2 \quad f^o_3(x) = s_3$
Step 2. Normalize all 1-class SVM scores using EVT models

\[ \lambda_{o,0}, \nu_{o,0}, \kappa_{o,0} \quad \lambda_{o,1}, \nu_{o,1}, \kappa_{o,1} \]

\[ \int_{s_0} \quad \int_{s_1} \]

Apply CDF for each class to each score

Probability model for test instance: \( P_o \)

\[ \lambda_{o,2}, \nu_{o,2}, \kappa_{o,2} \quad \lambda_{o,3}, \nu_{o,3}, \kappa_{o,3} \]

\[ \int_{s_2} \quad \int_{s_3} \]
Step 3: Test probabilities

\[ P_o(0|x) < \delta_{\tau}, \; \iota_0 = 0; \quad P_o(1|x) < \delta_{\tau}, \; \iota_1 = 0; \]
\[ P_o(2|x) < \delta_{\tau}, \; \iota_2 = 0; \quad P_o(3|x) < \delta_{\tau}, \; \iota_3 = 0 \]
Step 4: Apply indicator variables to binary SVMs

\[ P\eta,0(x) \times P\psi,0(x) \times \iota_0 = 0 < \delta_R \]
\[ P\eta,1(x) \times P\psi,1(x) \times \iota_1 = 0 < \delta_R \]
\[ P\eta,2(x) \times P\psi,2(x) \times \iota_2 = 0 < \delta_R \]
\[ P\eta,3(x) \times P\psi,3(x) \times \iota_3 = 0 < \delta_R \]
Models for class ‘3’ and the data point for this example

- Monotonically decreasing prob. bound
- Threshold on prob.
- Prob. from kernel machine varies locally with distance to training points
- W-SVM thresholded region
- CAP thresholded region

\[ P(3|Q) = 0 \]
W-SVM Implementation

Patch to LIBSVM available at:
https://github.com/ljain2/libsvm-openset

options:
-s svm_type : set type of SVM (default 0)
  0 -- C-SVC
  1 -- nu-SVC
  2 -- one-class SVM
  3 -- epsilon-SVR
  4 -- nu-SVR
  5 -- open-set oneclass SVM (open_set_training_file required)
  6 -- open-set pair-wise SVM (open_set_training_file required)
  7 -- open-set binary SVM (open_set_training_file required)
  8 -- **one-vs-rest WSVM** (open_set_training_file required)
  9 -- One-class PI-OSVM (open_set_training_file required)
  10 -- one-vs-all PI-SVM (open_set_training_file required)
Specialized Support Vector Machine (SSVM)


Boat dataset with 3 classes: red (the central class to the left), green (the central class to the right), and blue (the class with the ring shape).

(a) Class 1 (red). $b = -0.832$.

(b) Class 2 (green). $b = -0.86$.

(c) Class 3 (blue). $b = +0.594$. 
Specialized Support Vector Machine (SSVM)

Ensure bounded positively labeled open space by using an RBF kernel and **forcing the bias to be negative**

\[
b' \in \left\{ -\frac{|b|(2^i - 1)}{2|b| - 1}, i \in (0, |b|) \right\},
\]

Determined via open set grid search procedure
Specialized Support Vector Machine (SSVM)
Decision Boundary Carving

- Models the decision space of a kernelized (RBF) or linear SVM
  - Similar to 1-vs-Set Machine; move via error estimated from known negatives

- Largely domain specific (designed for camera source attribution and linking)

---

Feature extraction
Open set classification

Decision Boundary Carving
Open set classification

Decision Boundary Carving
Algorithm 1 Decision Boundary Carving (DBC)

1: Input: $\mathcal{P}, \mathcal{N} \quad \triangleright$ Set of elements of the positive class of interest and the known negative class(es), respectively
2: Output: Decision hyperplane parameters $\hat{w}$ and $b$, and the open set threshold $\varepsilon$
3: $$(\hat{w}, b) \leftarrow \text{SVM-Training}(\mathcal{P}, \mathcal{N});$$  \quad \triangleright \text{Calculating the SVM hyperplane parameters}
4: $C \leftarrow \text{Classification}(\hat{w}, b, \mathcal{P}, \mathcal{N});$ \quad \triangleright \text{Obtaining the decision scores}
5: min $\leftarrow \text{lowest-decision-score}(C);$ \quad \triangleright \text{Setting the initial data error to a maximum value}
6: max $\leftarrow \text{highest-decision-score}(C);$ \quad \triangleright \varepsilon'$ spans possible scores in C (increments of $10^{-4}$ herein)
7: $D \leftarrow +\infty$
8: For $\varepsilon' \leftarrow \text{min} \to \text{max}$ do
9: \hspace{1em} $(\Lambda^+, \Lambda^-) \leftarrow (0, 0)$
10: \hspace{2em} For all $x^+ \in \mathcal{P}$ do
11: \hspace{3em} $\Lambda^+ \leftarrow \Lambda^+ + \theta(x^+, \varepsilon')$; \quad \triangleright \text{True positives for this particular position of the hyperplane}
12: \hspace{2em} End For
13: \hspace{2em} For all $x^- \in \mathcal{K}$ do
14: \hspace{3em} $\Lambda^- \leftarrow \Lambda^- + \omega(x^-, \varepsilon')$; \quad \triangleright \text{True negatives for this particular position of the hyperplane}
15: \hspace{2em} End For
16: \hspace{2em} $A_x \leftarrow \frac{1}{2} \left( \frac{\Lambda^+}{|\mathcal{P}|}, \frac{\Lambda^-}{|\mathcal{K}|} \right);$ \quad \triangleright \text{Normalized averaged accuracy}
17: $D' \leftarrow \frac{1}{A_x}$;
18: If $D' < D$ then
19: \hspace{1em} $D \leftarrow D'$;
20: \hspace{1em} $\varepsilon \leftarrow \varepsilon'$;
21: \hspace{1em} End If
22: End For
23: Return $(\hat{w}, b, \varepsilon)$
Part 3: Case studies related to visual computing and other areas
Scene Analysis for Surveillance and Forensics

Image credit: David Green
Construct a “be on the lookout”

Suspect #2

- Wearing hat
- Male
- Not wearing sunglasses
- No beard
Search for common attributes across images

“Find males wearing hats, without beards or sunglasses”

Male: 0.62
Hat: 0.77
No beard: 0.60
Sunglasses: -0.36

Male: 0.77
Hat: 0.362
No beard: -0.02
Sunglasses: -0.46

Male: 0.90
Hat: 0.77
No beard: 0.69
Sunglasses: -0.60

Male: 0.70
Hat: 0.81
No beard: 0.40
Sunglasses: -0.03
Score Distributions

distribution of similarity scores for **different** objects

distribution of similarity scores for **same** objects

Impostors

d
false non-match

(g)
false match

non-match

decision threshold

\( \tau \)

match

Image Credit: A. Czajka
Accuracy as a statistic for open set problems

\[
\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}
\]

Imagine the following case:

1/100 \(TP\) correct
100,000/100,000 \(TN\) correct

99.9\% accuracy!
F-measure as a statistic for open set problems

Consistent point of comparison across inconsistent precision and recall numbers:

\[
F\text{-}measure = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}
\]
Open Set Object Recognition

Cross-data set methodology*
Training: Caltech 256

Testing: Caltech 256 + ImageNet

Open Universe of 88 classes: 1 positive class, \( n \) training classes, 87 negative testing classes (532,400 images)

Open Universe of 212 classes: 1 positive class, \( n \) training classes, 211 negative testing classes (13,610,400 images)

Features

Histogram of Oriented Gradients


LBP-like descriptor

# 1-vs-Set Machine vs. Typical SVMs

<table>
<thead>
<tr>
<th>2-tailed paired t-test</th>
<th>binary 1-vs-Set</th>
<th>binary linear</th>
<th>binary RBF</th>
<th>1-class 1-vs-Set</th>
<th>1-class linear</th>
<th>1-class RBF</th>
</tr>
</thead>
<tbody>
<tr>
<td>binary 1-vs-Set (HOG 88)</td>
<td>**</td>
<td>**</td>
<td>**</td>
<td>**</td>
<td>**</td>
<td>**</td>
</tr>
<tr>
<td>binary linear (HOG 88)</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>++</td>
<td>++</td>
<td>++</td>
</tr>
<tr>
<td>binary RBF (HOG 88)</td>
<td>—</td>
<td>++</td>
<td>—</td>
<td>++</td>
<td>++</td>
<td>++</td>
</tr>
<tr>
<td>1-class 1-vs-Set (HOG 88)</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>**</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>1-class linear (HOG 88)</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>1-class RBF (HOG 88)</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>++</td>
</tr>
<tr>
<td>binary 1-vs-Set (HOG 212)</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>*</td>
<td>—</td>
</tr>
<tr>
<td>1-class 1-vs-Set (HOG 212)</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>*</td>
</tr>
<tr>
<td>binary 1-vs-Set (LBP-like 88)</td>
<td>**</td>
<td>**</td>
<td>**</td>
<td>**</td>
<td>**</td>
<td>**</td>
</tr>
<tr>
<td>1-class 1-vs-Set (LBP-like 88)</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>binary 1-vs-Set (LBP-like 212)</td>
<td>—</td>
<td>—</td>
<td>**</td>
<td>—</td>
<td>**</td>
<td>**</td>
</tr>
<tr>
<td>1-class 1-vs-Set (LBP-like 212)</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
</tbody>
</table>

** 1-vs-Set Machine is statistically significant at p < 0.01

* 1-vs-Set Machine is statistically significant at p < 0.05

++ Baseline Machine is statistically significant at p < 0.01

— No statistical significance
Top 25 classes for the open universe of 88 classes
Top 25 classes for the open universe of 88 classes
F-measure as a function of openness
Near and far plane pressures for open universe of 88 classes

The second plane has an impact on recognition performance.
Biometric Verification

Does this incoming sample match the one in our system?

Answer: Verified or Not Verified
Open Set Face Verification

Labeled Faces in the Wild

Genuine Pair

Impostor Pair

Impostor Pair

Impostor Pair

Gallery classes: 12 people with at least 50 images
Impostor classes: 82 other people in LFW
1,316 test images across all classes
Features: LBP-like and Gabor*

Open set face verification

![Graph showing F-measure vs openness for different methods: Binary 1-vs-Set Machine, LBP-like, Binary SVM, LBP-like, Binary 1-vs-Set Machine, Gabor, and Binary SVM, Gabor. The graph displays a decrease in F-measure as openness increases.]
\( P_I \)-SVM Object Recognition
$P_I$-SVM Object Recognition

![Graph showing F-Measure vs Openness for different SVM variations](image-url)
Machine Learning Benchmark: LETTER

![Graph showing F-Measure vs Openness for different machine learning algorithms.]

- $P_{i}$-SVM
- $P_{i}$-OSVM
- MAS Thresh.
- 1-vs-Rest Mult. RBF Thresh.
- Pairwise Mult. RBF Thresh.
- 1-vs-Rest Mult. RBF
- Pairwise Mult. RBF
- Logistic Regression Thresh.
Machine Learning Benchmark: LETTER

![Graph showing F-Measure vs Openness for different models.](image-url)
Machine Learning Benchmark: LETTER

Accuracy vs. Openness for various classifiers:
- $P_I$-SVM
- $P_I$-OSVM
- MAS Thresh.
- 1-vs-Rest Mult. RBF Thresh.
- Pairwise Mult. RBF Thresh.
- 1-vs-Rest Mult. RBF
- Pairwise Mult. RBF
- Logistic Regression Thresh.
- One-Class RBF
Alternate Priors: Freq. of Occurrence of Letters in a Reference Corpus

![Graph showing F-Measure against Openness for different methods: P1-SVM, P1-OSVM, PI-SVM, PI-OSVM, MAS Thresh., 1-vs-Rest Mult. RBF Thresh., Pairwise Mult. RBF Thresh., Logistic Regression Thresh., One-Class RBF.](image)
W-SVM Object Recognition

$P_i$-SVM

F-measure vs. Openness

- W-SVM
- 1-vs-Set Machine
- MAS+CAP
- MAS
- 1-vs-All Bin. RBF
- 1-vs-All Bin. Lin.
Machine Learning Benchmark: LETTER

![Graph showing F-Measure vs. Openness for different machine learning models: W-SVM, MAS, 1-vs-All Mult. RBF Platt, Pairwise Mult. RBF, Logistic Regression, W-SVM δ, MAS+CAP, NN+CAP, NN, 1-vs-All-Mult. RBF. Each model is represented by a distinct marker and line style in the graph.](image)

- **W-SVM**: Blue square markers and line.
- **MAS**: Red cross markers and line.
- **1-vs-All Mult. RBF Platt**: Green triangle markers and line.
- **Pairwise Mult. RBF**: Orange diamond markers and line.
- **Logistic Regression**: Blue square markers and line.
- **W-SVM δ = .1**: Red circle markers and line.
- **MAS+CAP**: Green triangle markers with error bars.
- **NN+CAP**: Yellow square markers with error bars.
- **NN**: Orange diamond markers with error bars.
- **1-vs-All-Mult. RBF**: Grey triangle markers with error bars.
Fingerprint Spoof Detection

Incomplete knowledge of fabrication materials is always present at training time.
Materials and Quality

![Box plot showing quality comparison of different materials.](image)
Automatic detection and adaptation of a spoof detector to new spoof materials

W-SVM Novel Material Detector

Known Class: Live

Known Class: Latex

Known Class: Gelatine

Novel Materials in Open Space

1-Class Decision Boundary

Binary Decision Boundary
W-SVM Spoof Detector

1-Class Decision Boundary

Known Positive Class: Live

Known Negative Material: Gelatine

Known Negative Material: Latex

Known and Unknown Materials in Open Space
Experimental assessment of W-SVM

**Training:** LivDet 2011 is partitioned into 1,000 live and 400 spoof images corresponding to two fabrication materials.

**Testing:** LivDet 2011 is partitioned into two non-overlapping partitions $T_1$ and $T_2$.

Each $T_i$ consists of 500 live and 500 spoof images.

200 images are from spoof materials known at training time; 300 are from novel materials.

http://people.clarkson.edu/projects/biosal/fingerprint/
Performance difference between known and novel materials

<table>
<thead>
<tr>
<th>Training materials</th>
<th>( \mathcal{L}_{BSIF} )</th>
<th>( \mathcal{L}_{LBP} )</th>
<th>( \mathcal{L}_{LPQ} )</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>( EER_{\text{known}} ) [%]</td>
<td>( EER_{\text{novel}} ) [%]</td>
<td>( EER_{\text{known}} ) [%]</td>
<td>( EER_{\text{novel}} ) [%]</td>
</tr>
<tr>
<td>Skin+Latex+EcoFlex</td>
<td>6.0</td>
<td>16.3</td>
<td>6.5</td>
<td>13.2</td>
</tr>
<tr>
<td>Skin+WoodGlue+Latex</td>
<td>15.0</td>
<td>15.0</td>
<td>10.0</td>
<td>13.8</td>
</tr>
<tr>
<td>Skin+Gelatine+Latex</td>
<td>11.0</td>
<td>16.5</td>
<td>12.0</td>
<td>11.2</td>
</tr>
<tr>
<td>Skin+Silgum+Latex</td>
<td>10.5</td>
<td>20.8</td>
<td>12.3</td>
<td>19.7</td>
</tr>
<tr>
<td>Skin+EcoFlex+Silgum</td>
<td>14.0</td>
<td>29.5</td>
<td>9.3</td>
<td>30.2</td>
</tr>
<tr>
<td>Skin+Gelatine+EcoFlex</td>
<td>13.3</td>
<td>23.3</td>
<td>9.7</td>
<td>15.2</td>
</tr>
<tr>
<td>Skin+Silgum+Gelatine</td>
<td>13.3</td>
<td>23.8</td>
<td>11.5</td>
<td>23.3</td>
</tr>
<tr>
<td>Skin+WoodGlue+Silgum</td>
<td>18.3</td>
<td>23.0</td>
<td>18.0</td>
<td>32.3</td>
</tr>
<tr>
<td>Skin+Gelatine+WoodGlue</td>
<td>16.8</td>
<td>17.2</td>
<td>12.3</td>
<td>11.0</td>
</tr>
<tr>
<td>Skin+WoodGlue+EcoFlex</td>
<td>16.3</td>
<td>17.2</td>
<td>21.7</td>
<td>26.7</td>
</tr>
<tr>
<td><strong>Average EER ± STDERR</strong></td>
<td><strong>13.5 ± 1.1</strong></td>
<td><strong>20.3 ± 1.5</strong></td>
<td><strong>12.3 ± 1.4</strong></td>
<td><strong>19.7 ± 2.5</strong></td>
</tr>
</tbody>
</table>
Performance by feature set

<table>
<thead>
<tr>
<th>Texture descriptors used</th>
<th>EER(_M) ± STDERR [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Biometrika</td>
</tr>
<tr>
<td>Grey Level Co-occurrence Matrix (GLCM) [16]</td>
<td>44.6 ± 1.7</td>
</tr>
<tr>
<td>Binary Statistical Image Features (BSIF) [11]</td>
<td>33.2 ± 1.2</td>
</tr>
<tr>
<td>Local Phase Quantization (LPQ) [13]</td>
<td>34.3 ± 1.3</td>
</tr>
<tr>
<td>Binary Gabor Patterns (BGP) [50]</td>
<td>30.3 ± 1.0</td>
</tr>
<tr>
<td>Local Binary Patterns (LBP) [32]</td>
<td>32.5± 2.0</td>
</tr>
<tr>
<td>Local Binary Patterns (LBP) + Binary Gabor Patterns (BGP)</td>
<td>28.5 ± 1.2</td>
</tr>
</tbody>
</table>
Adapted spoof detector

<table>
<thead>
<tr>
<th>Training materials</th>
<th>Tested on $T_2$</th>
<th>Tested on $T_1$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\mathcal{L}^{LBP}$</td>
<td>$\mathcal{L}^{LBP'}$</td>
</tr>
<tr>
<td></td>
<td>(not adapted) [%]</td>
<td>(adapted using $T_1$) [%]</td>
</tr>
<tr>
<td>Skin+Latex+EcoFlex</td>
<td>14.6</td>
<td>13.4</td>
</tr>
<tr>
<td>Skin+WoodGlue+Latex</td>
<td>12.8</td>
<td>9.6</td>
</tr>
<tr>
<td>Skin+Gelatine+Latex</td>
<td>13.8</td>
<td>13.4</td>
</tr>
<tr>
<td>Skin+Silgum+Latex</td>
<td>18.2</td>
<td>14.0</td>
</tr>
<tr>
<td>Skin+EcoFlex+Silgum</td>
<td>29.6</td>
<td>18.0</td>
</tr>
<tr>
<td>Skin+Gelatine+EcoFlex</td>
<td>15.2</td>
<td>14.2</td>
</tr>
<tr>
<td>Skin+Silgum+Gelatine</td>
<td>22.2</td>
<td>15.8</td>
</tr>
<tr>
<td>Skin+WoodGlue+Silgum</td>
<td>30.4</td>
<td>14.4</td>
</tr>
<tr>
<td>Skin+Gelatine+WoodGlue</td>
<td>12.2</td>
<td>10.8</td>
</tr>
<tr>
<td>Skin+WoodGlue+EcoFlex</td>
<td>19.8</td>
<td>12.8</td>
</tr>
</tbody>
</table>

**Average EER ± STDERR:**

- Tested on $T_2$: $18.9 \pm 2.1$ (Skin+Latex+EcoFlex), $13.6 \pm 0.7$ (Skin+WoodGlue+EcoFlex)
- Tested on $T_1$: $14.0 \pm 2.0$ (Skin+Latex+EcoFlex), $7.7 \pm 0.5$ (Skin+WoodGlue+EcoFlex)
DET curves shift to the left after adaptation

Better
How well could you do with these features and the W-SVM?

<table>
<thead>
<tr>
<th>Sensors</th>
<th>Tested on $T_2$</th>
<th>Tested on $T_1$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(not adapted) [%]</td>
<td>(adapted using $T_1$) [%]</td>
</tr>
<tr>
<td>Biometrika</td>
<td>$L^{LBP}$</td>
<td>$L^{LBP'}$</td>
</tr>
<tr>
<td>Average EER ± STDERRROR:</td>
<td>18.9 ± 2.1</td>
<td>13.5 ± 0.6</td>
</tr>
<tr>
<td>Average EER ± STDERRROR:</td>
<td>$L^{LPQ}$</td>
<td>$L^{LPQ'}$</td>
</tr>
<tr>
<td>Average EER ± STDERRROR:</td>
<td>20.3 ± 0.5</td>
<td>14.6 ± 0.5</td>
</tr>
<tr>
<td>Average EER ± STDERRROR:</td>
<td>$L^{BSIF}$</td>
<td>$L^{BSIF'}$</td>
</tr>
<tr>
<td>Average EER ± STDERRROR:</td>
<td>21.5 ± 1.3</td>
<td>15.4 ± 0.6</td>
</tr>
</tbody>
</table>
Machine Learning Benchmark: LETTER

![Graph showing F-Measure vs. Openness for various models]

- **W-SVM**
- **MAS**
- **1-vs-All Mult. RBF Platt**
- **Pairwise Mult. RBF**
- **Pairwise Mult. RBF**
- **Logistic Regression**
- **W-SVM δ, = .1**
- **NN+CAP**
- **NN**
- **MAS+CAP**
- **1-vs-All-Mult. RBF**
Fingerprint Spoof Detection

Incomplete knowledge of fabrication materials is always present at training time.
Materials and Quality

Live
EcoFlex
Latex
Gelatine
WoodGlue
Silgum

Quality

35
40
45
50
55
60
65
Automatic detection and adaptation of a spoof detector to new spoof materials

Open Set Fingerprint Spoof Detection

Known or Unknown Material: Spoof

Acquired Fingerprints

Known Material: Live

Adapt Using Novel Spoof Materials

Binary W-SVM Spoof Detector

Decision: Live / Spoof

Gelatine

Latex

Novel Material

Live

Multi-class W-SVM Novel Material Detector

W-SVM Novel Material Detector

Known Class:
- Live
- Gelatine
- Latex

1-Class Decision Boundary
Binary Decision Boundary

Novel Materials in Open Space
W-SVM Spoof Detector

- Known Positive Class: Live
- Known Negative Material: Gelatine
- Known Negative Material: Latex
- Known and Unknown Materials in Open Space

1-Class Decision Boundary
Binary Decision Boundary
Experimental assessment of W-SVM

**Training:** LivDet 2011 is partitioned into 1,000 live and 400 spoof images corresponding to two fabrication materials.

**Testing:** LivDet 2011 is partitioned into two non-overlapping partitions $T_1$ and $T_2$.

Each $T_i$ consists of 500 live and 500 spoof images.

200 images are from spoof materials known at training time; 300 are from novel materials.
Performance difference between known and novel materials

<table>
<thead>
<tr>
<th>Training materials</th>
<th>$\mathcal{L}^{BSIF}$</th>
<th>$\mathcal{L}^{LBP}$</th>
<th>$\mathcal{L}^{LPQ}$</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\text{EER}_{\text{known}}$ [$%$]</td>
<td>$\text{EER}_{\text{novel}}$ [$%$]</td>
<td>$\text{EER}_{\text{known}}$ [$%$]</td>
<td>$\text{EER}_{\text{novel}}$ [$%$]</td>
</tr>
<tr>
<td>Skin+Latex+EcoFlex</td>
<td>6.0</td>
<td>16.3</td>
<td>6.5</td>
<td>13.2</td>
</tr>
<tr>
<td>Skin+WoodGlue+Latex</td>
<td>15.0</td>
<td>15.0</td>
<td>10.0</td>
<td>13.8</td>
</tr>
<tr>
<td>Skin+Gelatine+Latex</td>
<td>11.0</td>
<td>16.5</td>
<td>12.0</td>
<td>11.2</td>
</tr>
<tr>
<td>Skin+Sil gum+Latex</td>
<td>10.5</td>
<td>20.8</td>
<td>12.3</td>
<td>19.7</td>
</tr>
<tr>
<td>Skin+Eco Flex+Sil gum</td>
<td>14.0</td>
<td>29.5</td>
<td>9.3</td>
<td>30.2</td>
</tr>
<tr>
<td>Skin+Gelatine+Eco Flex</td>
<td>13.3</td>
<td>23.3</td>
<td>9.7</td>
<td>15.2</td>
</tr>
<tr>
<td>Skin+Sil gum+Gelatine</td>
<td>13.3</td>
<td>23.8</td>
<td>11.5</td>
<td>23.3</td>
</tr>
<tr>
<td>Skin+WoodGlue+Sil gum</td>
<td>18.3</td>
<td>23.0</td>
<td>18.0</td>
<td>32.3</td>
</tr>
<tr>
<td>Skin+Gelatine+Wood Glue</td>
<td>16.8</td>
<td>17.2</td>
<td>12.3</td>
<td>11.0</td>
</tr>
<tr>
<td>Skin+Wood Glue+Eco Flex</td>
<td>16.3</td>
<td>17.2</td>
<td>21.7</td>
<td>26.7</td>
</tr>
</tbody>
</table>

Average EER ± STDERR: 13.5 ± 1.1 20.3 ± 1.5 12.3 ± 1.4 19.7 ± 2.5 13.2 ± 0.9 18.8 ± 0.7 12.9 ± 1.0 19.6 ± 1.4
Performance by feature set

<table>
<thead>
<tr>
<th>Texture descriptors used</th>
<th>EER$_M$ ± STDERR [$%$]</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Biometrika</td>
</tr>
<tr>
<td>Grey Level Co-occurrence Matrix (GLCM) [16]</td>
<td>44.6 ± 1.7</td>
</tr>
<tr>
<td>Binary Statistical Image Features (BSIF) [11]</td>
<td>33.2 ± 1.2</td>
</tr>
<tr>
<td>Local Phase Quantization (LPQ) [13]</td>
<td>34.3 ± 1.3</td>
</tr>
<tr>
<td>Binary Gabor Patterns (BGP) [50]</td>
<td>30.3 ± 1.0</td>
</tr>
<tr>
<td>Local Binary Patterns (LBP) [32]</td>
<td>32.5 ± 2.0</td>
</tr>
<tr>
<td>Local Binary Patterns (LBP) +</td>
<td></td>
</tr>
<tr>
<td>Binary Gabor Patterns (BGP)</td>
<td>28.5 ± 1.2</td>
</tr>
</tbody>
</table>
Adapted spoof detector

<table>
<thead>
<tr>
<th>Training materials</th>
<th>Tested on $T_2$</th>
<th>Tested on $T_1$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\mathcal{L}^{LBP}$ (not adapted) [%]</td>
<td>$\mathcal{L}^{LBP'}$ (adapted using $T_1$) [%]</td>
</tr>
<tr>
<td>Skin+Latex+EcoFlex</td>
<td>14.6</td>
<td>13.4</td>
</tr>
<tr>
<td>Skin+WoodGlue+Latex</td>
<td>12.8</td>
<td>9.6</td>
</tr>
<tr>
<td>Skin+Gelatine+Latex</td>
<td>13.8</td>
<td>13.4</td>
</tr>
<tr>
<td>Skin+Silgum+Latex</td>
<td>18.2</td>
<td>14.0</td>
</tr>
<tr>
<td>Skin+EcoFlex+Silgum</td>
<td>29.6</td>
<td>18.0</td>
</tr>
<tr>
<td>Skin+Gelatine+EcoFlex</td>
<td>15.2</td>
<td>14.2</td>
</tr>
<tr>
<td>Skin+Silgum+Gelatine</td>
<td>22.2</td>
<td>15.8</td>
</tr>
<tr>
<td>Skin+WoodGlue+Silgum</td>
<td>30.4</td>
<td>14.4</td>
</tr>
<tr>
<td>Skin+Gelatine+WoodGlue</td>
<td>12.2</td>
<td>10.8</td>
</tr>
<tr>
<td>Skin+WoodGlue+EcoFlex</td>
<td>19.8</td>
<td>12.8</td>
</tr>
</tbody>
</table>

Average EER ± STDERR: 18.9 ± 2.1 13.6 ± 0.7 14.0 ± 2.0 7.7 ± 0.5
DET curves shift to the left after adaptation.
How well could you do with these features and the W-SVM?

<table>
<thead>
<tr>
<th>Sensors</th>
<th>Tested on $T_2$ (not adapted [%])</th>
<th>(adapted using $T_1$ [%])</th>
<th>Tested on $T_1$ (not adapted [%])</th>
<th>(adapted using $T_2$ [%])</th>
</tr>
</thead>
<tbody>
<tr>
<td>Biometrika</td>
<td>$L^{LBP}$</td>
<td>$L^{LBP'}$</td>
<td>$L^{LBP}$</td>
<td>$L^{LBP'}$</td>
</tr>
<tr>
<td>Average EER STDERROR:</td>
<td>18.9 ± 2.1</td>
<td>13.5 ± 0.6</td>
<td>14.0 ± 2.0</td>
<td>7.7 ± 0.4</td>
</tr>
<tr>
<td>Average EER ± STDERROR:</td>
<td>$L^{LPQ}$</td>
<td>$L^{LPQ'}$</td>
<td>$L^{LPQ}$</td>
<td>$L^{LPQ'}$</td>
</tr>
<tr>
<td>20.3 ± 0.5</td>
<td>14.6 ± 0.5</td>
<td>12.5 ± 0.7</td>
<td>9.0 ± 0.5</td>
<td></td>
</tr>
<tr>
<td>Average EER ± STDERROR:</td>
<td>$L^{BSIF}$</td>
<td>$L^{BSIF'}$</td>
<td>$L^{BSIF}$</td>
<td>$L^{BSIF'}$</td>
</tr>
<tr>
<td>21.5 ± 1.3</td>
<td>15.4 ± 0.6</td>
<td>13.1 ± 0.9</td>
<td>7.0 ± 0.4</td>
<td></td>
</tr>
</tbody>
</table>
Open Set Camera Attribution

Known Cameras

Unknown Cameras

1 2 3

x y z

...
Open Set Device Linking

Known Cameras

1 2 3

Unknown Cameras

x y z

...
Decision Boundary Carving: Dataset

- 13,210 images from 400 cameras (classes)
  - 25 known cameras (4,411 images)
  - 375 unknown cameras (8,799 images from Flickr) not processed in any form
Experiment

• We have access to sets of 15, 10, 5 and 2 suspect cameras from the 25 known cameras for training

• Tests with all the 400 cameras

• 5-fold cross validation

• Relative Accuracy: \( \frac{TP + TN}{2} \)
# Camera Attribution Results

<table>
<thead>
<tr>
<th></th>
<th>Open Set Cameras - Results in %</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>15</td>
</tr>
<tr>
<td><strong>Lukas et al. (2006)</strong></td>
<td></td>
</tr>
<tr>
<td>AVG</td>
<td>95.08</td>
</tr>
<tr>
<td>STD</td>
<td>2.40</td>
</tr>
<tr>
<td><strong>Li (2010)</strong></td>
<td></td>
</tr>
<tr>
<td>AVG</td>
<td>94.62</td>
</tr>
<tr>
<td>STD</td>
<td>2.56</td>
</tr>
<tr>
<td><strong>Our Approach</strong></td>
<td></td>
</tr>
<tr>
<td>(Only Central ROI)</td>
<td></td>
</tr>
<tr>
<td>AVG</td>
<td>90.95</td>
</tr>
<tr>
<td>STD</td>
<td>3.14</td>
</tr>
<tr>
<td>(Central ROI + DBC)</td>
<td></td>
</tr>
<tr>
<td>AVG</td>
<td>95.95</td>
</tr>
<tr>
<td>STD</td>
<td>1.70</td>
</tr>
<tr>
<td>(All ROIs)</td>
<td></td>
</tr>
<tr>
<td>AVG</td>
<td>95.75</td>
</tr>
<tr>
<td>STD</td>
<td>1.64</td>
</tr>
<tr>
<td>(All ROIs + DBC)</td>
<td></td>
</tr>
<tr>
<td>AVG</td>
<td>97.18</td>
</tr>
<tr>
<td>STD</td>
<td>1.63</td>
</tr>
</tbody>
</table>
Device Linking Results

<table>
<thead>
<tr>
<th>Exp. ID</th>
<th>Info. Used</th>
<th>Acc.</th>
<th>Std.Dev.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Goljan Baseline [3]</td>
<td>Original PRNU</td>
<td>51%</td>
<td>2.63</td>
</tr>
<tr>
<td>Goljan Baseline [3]</td>
<td>Enhanced PRNU</td>
<td>51.1%</td>
<td>2.14</td>
</tr>
<tr>
<td>Goljan ML Extended</td>
<td>Original PRNU</td>
<td>76.3</td>
<td>0.98</td>
</tr>
<tr>
<td>Goljan ML Extended</td>
<td>Enhanced PRNU</td>
<td>75.6%</td>
<td>0.97</td>
</tr>
<tr>
<td>All ROIs</td>
<td>Original PRNU</td>
<td>86.7</td>
<td>0.81</td>
</tr>
<tr>
<td>All ROIs</td>
<td>Enhanced PRNU</td>
<td>87.4%</td>
<td>1.62</td>
</tr>
<tr>
<td>All ROIs + DBC</td>
<td>Original PRNU</td>
<td>86.5</td>
<td>1.31</td>
</tr>
<tr>
<td>All ROIs + DBC</td>
<td>Enhanced PRNU</td>
<td>87.4%</td>
<td>2.51</td>
</tr>
</tbody>
</table>
We find open set problems in NLP too

An emerging area: **Stylometry**

Machine learning is now able to detect quantifiable style markers in written language.

Common applications: authorship attribution, genre tagging, **textual reuse**

No text exists in isolation: we cannot make *a priori* assumptions on characteristics of style.
Intertextuality in Literature

Kristeva: “Any text is constructed as a mosaic of quotations; any text is the absorption and transformation of another.”

The nature of textual reuse is widely varied:

- Direct quotations
- Loose lexical correspondance
- Idea reuse
- Sound reuse

Since the problem is one of pattern recognition, it is a good candidate for automated assistance by computers.

Quantitative Intertextuality is the algorithmic study of information reuse in any semiotic system.

Applications:

- Scholarly work (*e.g.* digital humanities)
- Practical applications (*e.g.* digital forensics)

Feature: the functional n-gram

A functional bi-gram is an n-gram-based feature that describes frequently appearing information.

Functional n-grams for **sound** are:

- Character-level features
- Stand-ins for phonemes
- Similar to function words
- Elements of most of the lexicon

\[
P(e_n|e_{n-N+1}^{n-1}) = \frac{C(e_{n-N+1}^{n-1} e_n)}{C(e_{n-N+1}^{n-1})} \iff \text{freq}(e_{n-N+1}^{n-1} e_n) > \phi
\]

The functional n-gram process

Select $x$ of the most frequently occurring n-grams in a sample:

| 804 | er | 560 | ti |
| 778 | qu | 555 | us |
| 726 | is | 513 | at |
| 723 | en | 512 | nt |
| 709 | re | 503 | ae |
| 685 | te | 501 | ta |
| 651 | es | 470 | tu |
| 615 | um | 468 | ri |
| 604 | in | 454 | or |
| 574 | it | 452 | am |

Compute the probability features:

\[
\begin{align*}
\text{Count(“er“)} &= \frac{804}{560 + 555 + 513 + 
\text{Count(“e“)} &= \frac{560}{560 + 555 + 513 + 
\text{Count(“re“)} &= \frac{804}{560 + 555 + 513 + 
\end{align*}
\]

\[
= 0.179785
\]

\[
= 0.275447
\]
Open Set Influence Analysis

We want to test the stylistic similarity of any author to another.

Methodology: train a 1-class SVM on representative samples from a known author.

Recall that 1-class SVMs tend to overfit the training data.

We don’t need to generalize - we are only interested in samples that fall within the support of the training data.
Open Set Influence Analysis
Historical “Forensics”

Samples classified positively with Catullus out of all samples:

<table>
<thead>
<tr>
<th>More Like Catullus</th>
<th>Text</th>
<th>Positive Class.</th>
<th>Less Like Catullus</th>
<th>Text</th>
<th>Positive Class.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Angustae Vitae</td>
<td>1/1</td>
<td>Ovid Amores</td>
<td>2/40</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Propertius Elegies</td>
<td>6/40</td>
<td>Horace Epistles</td>
<td>3/40</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tibullus Elegies</td>
<td>5/40</td>
<td>Virgil Aeneid</td>
<td>2/35</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Unknown attacks can appear amongst the myriad number of known attacks
Two detection techniques

1. Signature-based IDS
   - Compare packets against a database of signatures of known attacks
   - Unclear where a discriminative probability model would help here

2. Statistical anomaly-based IDS
   - Compare traffic to established baseline characterizing “normal” activity
   - Highly amenable to open set recognition techniques
Future directions in IDS

• Feature space: what will help us generalize to unseen data?

• Application of W-SVM

• Other probability models?
  - One-class modeling is popular for anomaly detection in many domains
Part 4: Research opportunities and trends
Research opportunities and trends

• The open set recognition problem and new feature characterization methods (e.g., deep learning)

• Integrating open set solutions with the image characterization process directly (strongly generalizable image characterization)

• Opportunities for novelty detection and automatic addition of classes (online adaptation)

• Bringing the user into the loop (relevance feedback)

• Final considerations
Open World Evaluation

Parameter Learning Phase

Incremental Learning Phase

Training phase

Testing phase

Closed Set Testing

Open Set Testing

Known Categories

Unknown Categories
Training for Open World

- Parameter Learning with initial set of categories
- Estimation of $\tau$ for open set learning to balance open space risk
- Optimize for Known vs Unknown Errors
- Incrementally add new categories

NCM - ML

NNO
Learning Novel Concepts

Nearest Class Mean Classifier

Nearest Non Outlier Algorithm

Adding Novel Concepts to the System
Opening Deep Networks

- Softmax always has a “winner” and re-weights scores
- Networks are easily fooled with high confidence
- “Fooling” images are obviously “open set” and should be rejected
- Adversarial images are more problematic - visually close but often far in label space

A. Bendale and T. Boult “Towards Open Set Deep Networks” CVPR 2016
Opening Deep Networks

Can hill climb to find fooling images*

Adversarial Manipulation of AlexNet

Hammerhead Image + Noise (*100)

These are “visually near” but mislabeled

AlexNet

Softmax Output (.32, ScubaDiver)

Adversarial images generated using: Goodfellow, Sheln and Szegedy “Explaining and harnessing adversarial examples,” ICLR 2015
MAV and OpenMax

- Insight: A class is represented not just by its output, but by its Mean Activation Layer (scores for all classes)
- MAV is just the average in penultimate layer
- “EVT distances” from MAV is a CAP model
- Given MAV, estimate probability of “unknown” via EVT and OpenMax = Softmax type normalized probability including probability of unknown
Open Set Deep Networks

Idealized class

Softmax Output (0.992, baseball)

Real: SM 0.94

Fooling: SM 1.0,

Openset: SM 0.15

MODEL
Real Image
Fooling
OpenSet

Baseball

Sharks
Whales
Dogs
Fish
Baseball
Real: SM 0.57, OM 0.58
Fooling: SM 0.98, OM 0.00
Openset: SM 0.25, OM 0.10
Adversarial Scuba Diver
SM 0.32 Scuba Diver
OM 0.49 Unknown
Open Set Deep Networks

Text

Real: SM 0.94
OM 0.94

Fooling: SM 1.0,
OM 0.00

Openset: 0.15,
OM: 0.17
Relevance Feedback

Credit: Manning et al. 2009
Wrapping up...
Further Reading


Further Reading


Code

1-vs-Set Machine, $P_l$-SVM, and W-SVM on GitHub: https://github.com/ljain2/libsvm-openset

Data sets:
http://www.metarecognition.com/openset/
The Open Set Recognition Problem

and Its Implications and Opportunities in Visual Computing, Forensics and Security

Anderson Rocha
University of Campinas, Brazil
anderson.rocha@ic.unicamp.br

Walter J. Scheirer
University of Notre Dame, U.S.A.
walter.scheirer@nd.edu