Image Clustering by Source Camera via Sparse Representation

**Quoc-Tin Phan**, Giulia Boato, Francesco G. B. De Natale

University of Trento, Italy.

Contact:
- Email: quoctin.phan@unitn.it
- Personal page: http://quoctin.github.io
All cameras (CMOS, CCD, etc.) have an intrinsic pattern noise: FPN + PRNU

PRNU properties: robustness, stability, universality.

Automatically suppressed by cameras

Pattern noise

Fixed Pattern Noise (FPN)

Photo-Response NonUniformity (PRNU)

Dominant component
IMAGE CLUSTERING BY SOURCE CAMERA
IMAGE CLUSTERING BY SOURCE CAMERA
IMAGE CLUSTERING BY SOURCE CAMERA
IMAGE CLUSTERING BY SOURCE CAMERA
IMAGE CLUSTERING BY SOURCE CAMERA
EXISTING CLUSTERING PIPELINE

Camera fingerprint estimation

Correlation calculation

Clustering

\[
W = I - \text{filter}(I)
\]

\[
W : \text{noise residual}
\]

Spectral clustering
Hierarchical clustering
Markov Random Field

Affinity matrix

\[
\begin{align*}
(W^{(i)} - \bar{W}^{(i)}1^T) \odot (W^{(j)} - \bar{W}^{(j)}1^T) \\
\|W^{(i)} - \bar{W}^{(i)}1^T\| \|W^{(j)} - \bar{W}^{(j)}1^T\|
\end{align*}
\]

\[
\overline{W} : \text{arithmetic mean}
\]
OUR CONTRIBUTION

Camera fingerprint estimation

$W = I - \text{filter}(I)$

$W$ : noise residual

Correlation calculation

Sparse Representation Learning

- Better represent relationship among data points in high dimensional space.
- Avoid pairwise correlation calculation.

Clustering

- Spectral clustering
- Hierarchical clustering
- Markov Random Field
OVERALL SCHEMA

Input data → Fingerprint Estimation → Sparse Rep. Learning → Spectral clustering
Sparse Subspace Clustering (SSC) was proposed by [E. Elhamifar et al. 2009] and theoretically analyzed in [E. Elhamifar et al. 2013].

Each data point can be expressed as the linear combination of a proper basis.

If the input data sufficiently fulfilled the underlying subspaces, the data matrix itself can be used as the subspace basis.
For each, \( X_{:,i} \) we want to learn its corresponding sparse representation \( Z_{:,i} \) by solving \( \ell_1 \)-regularized least squares.

\[
Z_{:,i} = \arg\min_z \gamma \|z\|_1 + \|X_{:,i} - X_{:,i}z - X_{:,i}\|_2^2
\]

\( \text{Sparsity} \quad \text{Reconstruction error} \)
SOLVING $\ell_1$-REGULARIZED LEAST SQUARES

- The objective function is non-smooth due to $\ell_1$ norm.

- The dimensionality of data is high, off-the-shelf optimization tools, e.g., CVX, cannot manage.

Alternating Direction Method of Multiplier (ADMM) [Boyd. 2011]

- Decouple non-differentiable term and differentiable term.

$$Z_{:,i} = \arg\min_z \gamma\|v\|_1 + \|X_{:,i} - X_{:,i} - X_{:,i}z\|^2_2 \quad \text{s.t.} \quad v = z$$

- At each iteration:
  - Keep $v$ fixed and update $z$ by an analytical solution.
  - Keep $z$ fixed and update $v$ by soft thresholding.
ESTIMATING NUMBER OF CLUSTERS

- **Step 1:** Generate the affinity matrix and its corresponding normalized Laplacian matrix

\[ H = |Z| + |Z^T| \]

\[ L = I - D^{-1/2}HD^{-1/2} \quad \text{where} \quad D_{i,i} = \sum_j^N H_{i,j} \]

- **Step 2:** Estimate the number of clusters \( \hat{K} \) using eigengap heuristic

\[ \hat{K} = \arg\max_K \lambda_{K+1} - \lambda_K, \]

where \( \lambda_1, \cdots, \lambda_N \) are non decreasing eigenvalues of \( L \)
ESTIMATING REGULARIZATION PARAMETER

\[ Z_{:,i} = \arg \min_z \gamma \|v\|_1 + \|X_{:,i} - X_{:,j}\|_2^2 \quad \text{s.t.} \quad v = z \]

- The regularization parameter \( \gamma \) controls the tradeoff between the sparseness of solutions and the reconstruction errors.
- **Too dense** solutions result in merged clusters.
- **Too sparse** solutions result in many small clusters.

On a development set, select \( \gamma \) minimizing Normalized Cuts and maximizing eigengap.
**EXPERIMENTAL SETUP**

- Images from **Dresden** [T. Gloe et al. 2010] and **RAISE** [D.-T. D.-Nguyen et al. 2015] are used to generate 5 datasets.

<table>
<thead>
<tr>
<th>Name</th>
<th>From</th>
<th>#models</th>
<th>#cameras</th>
<th>#images/camera</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>$D_1$</td>
<td>RAISE</td>
<td>3</td>
<td>3</td>
<td>50</td>
<td>150</td>
</tr>
<tr>
<td>$D_2$</td>
<td>Dresden</td>
<td>10</td>
<td>10</td>
<td>100</td>
<td>1000</td>
</tr>
<tr>
<td>$D_3$</td>
<td>RAISE</td>
<td>3</td>
<td>3</td>
<td>50, 75, 100</td>
<td>225</td>
</tr>
<tr>
<td>$D_4$</td>
<td>Dresden</td>
<td>10</td>
<td>10</td>
<td>40, 60, 80, 100</td>
<td>660</td>
</tr>
<tr>
<td>$D_{ev}$</td>
<td>Dresden</td>
<td>5</td>
<td>5</td>
<td>100</td>
<td>500</td>
</tr>
</tbody>
</table>

$D_1, D_2$ are symmetric; $D_3, D_4$ are asymmetric; $D_{ev}$ is used as the development set.
EXPERIMENTAL SETUP

- Evaluation metric:
  - F1-score.
  - The ratio of number of discovered clusters $K_p$ to the number of ground-truth clusters $K_g$.

- Comparison to SoA:
  - **MRF**: Markov Random Field (Li et al. 2010).
  - **MSC**: Multiclass Spectral Clustering (Liu et al. 2010).
  - **SCNcuts**: Spectral Clustering with Normalized Cut criterion (Amerini et al. 2014).
  - **HC**: Hierarchical Clustering (Villalba et al. 2015).
  - **LS**: Large-Scale method (Lin et al. 2017).
EXPERIMENTAL RESULTS

- Clustering results on symmetric datasets $D_1$, $D_2$ and asymmetric dataset $D_3$, $D_4$.

<table>
<thead>
<tr>
<th>Method</th>
<th>$D_1$</th>
<th>$D_2$</th>
<th>$D_3$</th>
<th>$D_4$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>F1</td>
<td>$K_P/K_G$</td>
<td>F1</td>
<td>$K_P/K_G$</td>
</tr>
<tr>
<td>HC</td>
<td>0.75</td>
<td>28/3</td>
<td>0.62</td>
<td>48/10</td>
</tr>
<tr>
<td>MSC</td>
<td>0.98</td>
<td>5/3</td>
<td>0.80</td>
<td>21.7/10</td>
</tr>
<tr>
<td>SCNCuts</td>
<td>0.99</td>
<td>3/3</td>
<td>0.32</td>
<td>2/10</td>
</tr>
<tr>
<td>MRF</td>
<td>0.82</td>
<td>9.6/3</td>
<td>0.81</td>
<td>24/10</td>
</tr>
<tr>
<td>LS</td>
<td>0.93</td>
<td>5/3</td>
<td>0.91</td>
<td>12/10</td>
</tr>
<tr>
<td>Proposed</td>
<td>0.97</td>
<td>3/3</td>
<td>0.91</td>
<td>11/10</td>
</tr>
</tbody>
</table>

Int. Workshop on Multimedia Forensics and Security, 2017
EXPERIMENTAL RESULTS

- Running time comparison to SoA methods.

- Normalized correlation calculation: built-in function $\text{corr2}$
CONCLUSION

- A complete clustering framework for image clustering by source camera.
  - Learn sparse representations of camera fingerprints by solving $\ell_1$-regularized least squares.
  - Estimate number of clusters (cameras).
  - Without the need to set regularization parameter.
  - Outperform SoA methods.
FUTURE WORK

- Validate the framework on cameras of the same model.
- Extend the framework to deal with large-scale datasets.
- Make the software publicly available.
REFERENCES


THANK YOU!!!

Q&A