Francesco Marra

Tutor: Carlo Sansone – co-Tutor: Luisa Verdoliva

XXX Cycle - II year presentation

Digital Image Forensics in the era of social networks

DIGITAL IMAGE FORENSICS

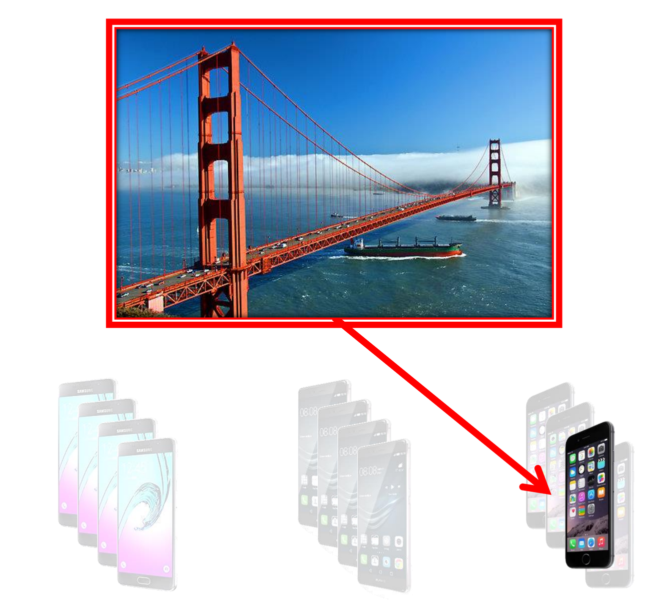
Digital Image Forensics deals with the authentication and the analysis of a digital image aimed at providing useful evidence before a court [1].
We broadly focus on the two following topics:

- **image integrity detection**: it establishes if a digital image has undergone malicious post-processing or tampering;
- **image source identification**: it determines whether an image is taken from a given camera or model.

Nowadays it is extremely easy to share digital information thanks to the use of **social networks**. In 2014 more than 1.8 billion images and videos have been published on the net each day. Besides its explicit value, this wealth of visual data represents a precious source of information for investigations. In order to detect illegal activities and combat crime it is important to determine the origin of a visual asset and its integrity. Likewise, proving that a photo was taken by a given camera or has not been manipulated may be important before a court of law.

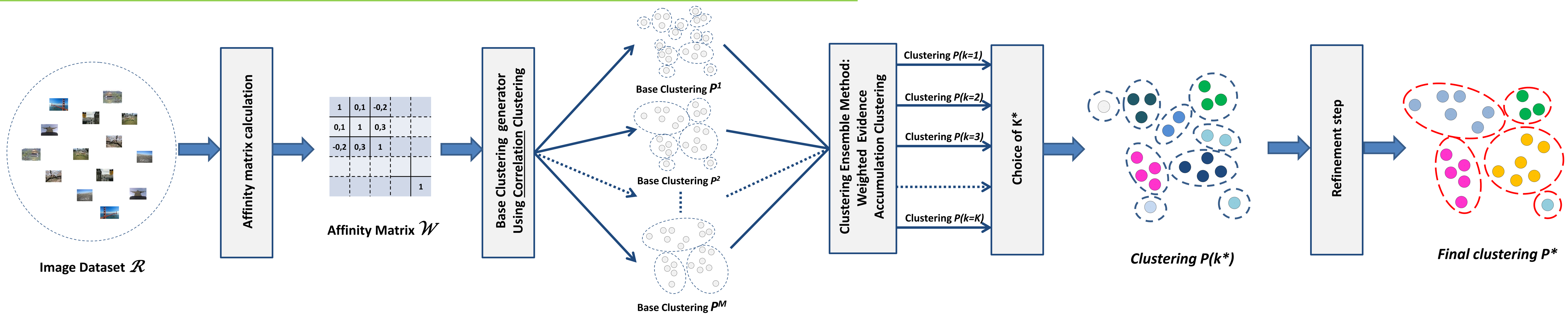
IMAGE SOURCE IDENTIFICATION

Which is the specific camera that has taken this photo?

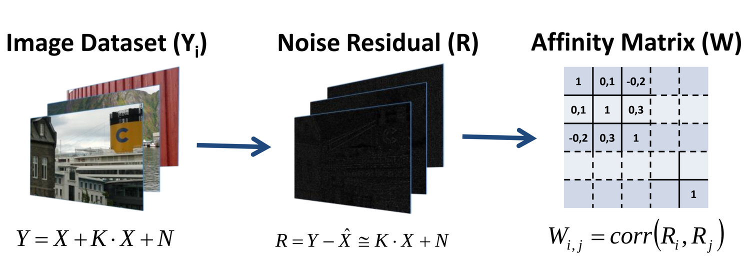


- **Image source identification** is typically based on the Photo-Response Non Uniformity (**PRNU**) noise of the camera
- PRNU-based methods assume that the camera PRNU pattern is known, or that a large number of images come from the target camera are available [2].
- These assumptions are not met **in real-world applications**, especially if we think of an open set scenario, like a social network. For this reason, we relax or remove this hypothesis, by developing a blind method.

PROPOSED METHOD: Blind Image Source Identification



PRNU-based Affinity Matrix



The correlation between images residuals coming from the same camera are higher than the ones coming different camera

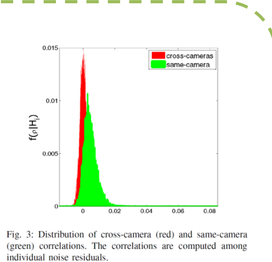


Fig. 3: Distribution of cross-camera (red) and same-camera (green) correlations. The correlations are computed among individual noise residuals.

Correlation clustering

Recasting the problem as a graph ***G=(V,E,W)*** partitioning where:

V: each node denotes an image
E: each edge denotes an image-image relationship
W: each edge weight measures pairwise similarity

$$x_e = \begin{cases} 1 & \text{Edge Cut} \\ 0 & \text{Edge Retained} \end{cases}$$

$$\mathbf{x} \in X_c \Leftrightarrow \begin{array}{l} \forall i,j,k \\ x_{ij} - x_{ik} - x_{jk} \leq 0 \end{array}$$

$$\varepsilon(\mathbf{x}) = \sum_{e \in E} w_e x_e$$

$$\mathbf{x}^{CC} = \underset{\mathbf{x} \in X_C}{\operatorname{argmin}}\, \varepsilon(\mathbf{x})$$

Scaling with an α factor:

$$W = W - \alpha$$

- So that, if $w_{i,j}>0$ i and j coming from the same camera/cluster
- Using multiple α we obtain multiple partitions for the next step

WEAC Ensemble Clustering

Running the Correlation Clustering with different parameter, we obtain *M base clustering* $\mathcal{P} = \{P^1, P^2, \dots, P^M\}$. For each partition P^i, we calculate the co-occurrence matrix ***S:***

$$S^i_{j,k} = \begin{cases} 1 & \text{if } P^i(j) = P^i(k) \\ 0 & \text{otherwise} \end{cases}$$

With the *M* similarity matrix we can obtain the Weighted Evidence Accumulation matrix of WEAC [11]:

$$A = \sum_{i=1}^{M} w_i S^i \qquad w_i = \frac{\mathrm{CAI}_i^\beta}{\sum_{j=1}^M \mathrm{CAI}_j^\beta} \qquad \mathrm{CAI}_i = \frac{1}{M-1} \sum_{j=1, j\neq i}^{M} \mathrm{sim}(P^i, P^j)$$

Using a Single-linkage clustering and calculating the objective ***k**** we obtain the desidered partition ***P(k*)***

MLE-based Clustering Refinement

Algorithm 1 Clustering Refinement

```
1:  procedure C = REFINEMENT({R}, {C})
2:      T_|C| = 0
3:      do
4:          changed = FALSE
5:          Increase T_|C|
6:          L = {C_p : |C_p| > T_|C|}
7:          S = {C_p : |C_p| ≤ T_|C|}
8:          compute K~_p for all C_p ∈ L
9:          compute Λ_pq for all C_p ∈ L and C_q ∈ S
10:         while max_(p,q) Λ_p,q > 0 do
11:             changed = TRUE
12:             (p*, q*) = arg max_(p,q) Λ_p,q
13:             C_s = merge(C_p*, C_q*)
14:             update L, S
15:             update K_p for all C_p ∈ L
16:             update Λ_pq for all C_p ∈ L and C_q ∈ S
17:         end while
18:     while changed
19: end procedure
```

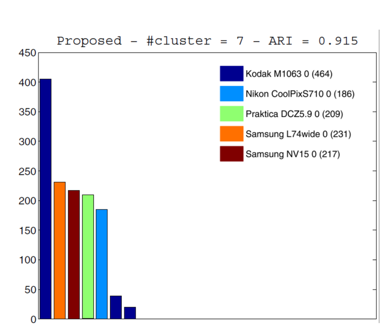
Experimental Results

Set	Models
Set A	I70, Z150, D200, μ, RCP
Set B	M1063, S710, DCZ, L74w, NV15
Set C	all ten models

TABLE II: Heterogeneous datasets used in the experiments For each dataset (*e.g.,* A) we consider three version, with one (A.1) two (A.2) or all (A.max) devices per model.

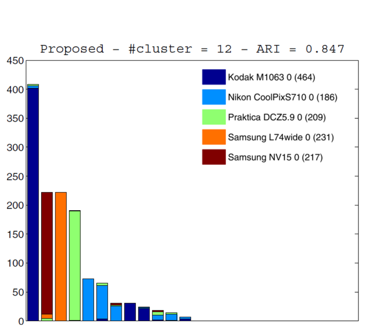
Set	#Dev	Bloy2008	Amerini2014	Fahmy2015	Marra2016	proposed	*Ncut oracle*	*CC oracle*
A.1	5	0.708	0.763	0.707	0.665	**0.916**	*0.872*	*0.908*
A.2	10	0.725	0.699	0.683	0.813	**0.852**	*0.801*	*0.848*
A.max	18	0.689	0.568	0.374	0.398	**0.729**	*0.665*	*0.761*
B.1	5	0.388	0.722	0.324	0.911	**0.915**	*0.722*	*0.736*
B.2	10	0.538	0.606	0.505	0.833	**0.836**	*0.683*	*0.819*
B.max	21	0.464	0.451	0.457	0.860	**0.881**	*0.631*	*0.834*
C.1	10	0.627	0.669	0.703	0.844	**0.865**	*0.669*	*0.836*
C.2	20	0.683	0.607	0.486	0.856	**0.956**	*0.607*	*0.929*
C.max	39	0.598	0.536	0.413	0.686	**0.821**	*0.536*	*0.798*

TABLE III: Performance on heterogeneous sets.



Set	#Dev	Bloy2008	Amerini2014	Fahmy2015	Marra2016	proposed	*Ncut oracle*	*CC oracle*
A.1	5	0.272	0.301	0.196	0.402	**0.723**	*0.305*	*0.513*
A.2	10	0.395	0.233	0.106	0.471	**0.592**	*0.294*	*0.567*
A.max	18	0.244	0.134	0.033	**0.538**	0.532	*0.171*	*0.569*
B.1	5	0.529	0.592	0.098	0.691	**0.847**	*0.705*	*0.792*
B.2	10	0.355	0.357	0.036	0.621	**0.718**	*0.394*	*0.683*
B.max	21	0.232	0.078	0.019	0.547	**0.644**	*0.084*	*0.597*
C.1	10	0.461	0.309	0.089	0.618	**0.646**	*0.338*	*0.643*
C.2	20	0.324	0.173	0.064	**0.573**	0.485	*0.203*	*0.636*
C.max	39	0.220	0.031	0.026	0.571	**0.601**	*0.049*	*0.589*

TABLE V: Performance on heterogeneous sets after high quality uploading on FaceBook.



Open issues

- (Big) Data
- Video
- Deep Learning

References

[1] R. Kaur, A. Kaur, "Digital Forensics", International Journal of Computer Applications V.50 – No.5, 2012

[2] J. Lukas, J. Fridrich, and M. Goljan, "Digital camera identification from sensor pattern noise", IEEE Transactions on Information Forensics and Security, vol. 1, no. 2, pp. 205–214, 2006.

[3] G. J. Bloy, "Blind camera fingerprinting and image clustering," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 30, no. 3, pp. 532–534, 2008.

[4] O. Fahmy, "An efficient clustering technique for cameras identification using sensor pattern noise," in International Conference on Systems, Signals and Image Processing, pp. 249–252 , 2015

[5] I. Amerini, R. Caldelli, P. Crescenzi, A. D. Mastio, and A. Marino, "Blind image clustering based on the normalized cuts criterion for camera identification", Signal Processing: Image Communication, vol. 29, no. 8, pp. 831 – 843, 2014

[6] J. Shi and J. Malik, "Normalized cuts and image segmentation", IEEE Transaction Pattern Analysis Machine Intelligence, vol. 22, no. 8, pp. 888–905, 2000.

[7] F. Marra, G. Poggi, C. Sansone, L. Verdoliva - "Camera model identification through SPAM features", Multimedia Tools and Applications, 2016

[8] N. Bansal, A. Blum, and S. Chawla, "Correlation clustering", Foundations of Computer Science. Proceedings, pp. 238–247, 2002

[9] S. Bagon and M. Galun, "Large scale correlation clustering optimization", CoRR, vol. abs/1112.2903, 2011.

[10] C. Rother, V. Kolmogorov, V. Lempitsky, and M. Szummer, "Optimizing binary MRFs via extended roof duality", IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–8 , 2007

[11] D. Huang and J.-H. Lai and C.-D. Wang, "Combining multiple clusterings via crowd agreement estimation and multi-granularity link analysis", Neurocomputing, 2015

In progress…

Are Device Fingerprinting techniques effective on Social Networks?