INVESTIGATING IMAGE DEPENDENCIES THROUGH IMAGE FORENSICS

*F. Uccheddu, A. De Rosa, A. Piva**

University of Florence
Dept. of Electronics and Telecommunications
Via S.Marta 3, 50139 - Firenze, Italy

M. Barni

University of Siena
Dept. of Information Engineering
Via Roma 56, 53100 - Siena, Italy

ABSTRACT

In many applications it would be useful to understand if, in a set of images representing a given scene, there is some pairwise relationship between them. In this paper a formalization of the above problem resulting in a dependency test and a dependency graph is presented and a roadmap for research in this direction is given. Moreover, a practical example based on an oversimplified case study with a basic implementation is shown, to demonstrate the feasibility of the proposed approach.

***Index Terms*—** image dependencies, image forensics, distance test, graph, ontology

1. INTRODUCTION

The large amount of information deriving from the Web in its multifaceted forms (e.g. text, sounds, images) trough varied instruments (e.g. web pages, blogs, news, wikipedia), has a pronounced effect on opinions and bias of every person using the Web. It is well known that an image can capture the attention of a viewer more than a long sentence ("a picture is worth a thousand words" is a famous cliche), hence the influence that images have on users cannot be underestimated. Starting from such a consideration, the knowledge of the origin of an image and the certainty of its authenticity, should serve as warranty that bias deriving from it are not malevolently directed.

The advent of technologies enabling modification and manipulation of digital images has urged the parallel advent of technologies for image analysis and content integrity verification. Even if the proposed (active and passive) technologies provide an interesting knowledge about image history and a consequent help for the comprehension of the opinion mining process related to image viewing, all such instruments only consider the analysis of single images. On the contrary, there is another type of analysis that could greatly improve the understanding of how images on the Web influence users: the exploration of image dependency. If it would be possible to know how a set of images are related one to each other (e.g. to know that one image is achieved by processing another image), we could be more aware of what we see.

Let us consider a common situation, where a person is interested in getting information from the Web on a specific topic; let us suppose that such information are in the form of visual data, i.e. a set of images. Opinion building is strongly influenced by viewing such images and in particular by knowing the relationship between them: in fact, one may wonder if these contents are all original, or most of them have been derived from just one. If all the images have been derived by a master content, hence we can state that this content had an important social/historical influence, whereas if all the documents have been independently created, hence all the images have their own importance in the generation of an idea.

By considering the mentioned scenario, our interest is to propose a possible approach for analyzing image dependency; in this paper we propose a possible formalization of the above ideas and a roadmap for research in this direction: after clarifying what finding image dependencies means for us, we propose a formalization of the presented problem resulting in a dependency test and a dependency graph, with the relative graph construction. In the second part of the paper, we give a practical example based on an oversimplified case study to highlight how the formalization introduced in the previous section can be implemented in practice.

2. DISCOVERING IMAGES HISTORY

Several approaches have been proposed in the last few years for studying digital images and capturing information about their history: starting from active approaches based on watermarking and digital signatures [1], through passive approaches typical of image forensics. When digital images had to be protected or their authenticity verified or, furthermore, their provenance tracked, the solution generally was to insert in the original data an embedded, usually unperceivable, information that permitted afterwards to determine what was happened, in which part of the content and, in particular application cases, by whom. This kind of techniques that can be grouped under the name of digital watermarking [2],

*The work described in this paper has been partially supported by the European Commission funded project LivingKnowledge.

follow an "active" approach, that is it is necessary to operate on the original document which has to be available from the beginning; this kind of active technologies [3] can be adopted to manage data in a specific application context where additional information casting is feasible but are not able to deal with an open operative environment in which only a detection step is possible. On the contrary, in this situation a passive methodology would be useful; with the term "passive" an approach which tries to make an assessment only having the digital content at disposal is to be intended. It is straightforward to realize that this kind of investigation is harder and has to be founded on the thorough analysis of some intrinsic features (fingerprints) that should have/have not been present and are not/are now recognizable inside the observed data [4]. For sake of clarity: when a photomontage, for instance, has been performed to alter the content of a digital photo, to change the meaning of the represented scene, some traces of this operation are left somehow over the new fake image. These traces, although unperceivable, can result in the modification of the image structure such as anomalous pixel values (e.g. sequential interpolated values or strange continuous flat values) but also in inconsistencies within the image content itself such as anomalies in the illumination direction or in the presence of slight disproportionate object size with respect to the whole context.

Specifically, a bunch of image forensics techniques have been developed that permit to extract knowledge about the origin of the content [5], or to detect the application of a wide variety of manipulations, including image resampling, single and double JPEG compression, and cut and paste and slicing operations [4, 6]. Though the current state of the art of both active and passive technologies permits to acquire very interesting information about *image history*, all the instruments developed so far focus on the analysis of single images. In several applications, though, the investigation of image dependencies may be of similar, or greater, importance. For instance, knowing how a set of images are related one to each other could allow the clustering of images sharing the same root image or images that followed a similar *history*. In this way, we could discover that the images regarding an event have been produced from a limited set of source images, thus permitting to isolate the original information brought by each image from the information common to all the other images in the set. In other situations, knowing how a few source images have evolved into a large set of derived pictures, could allow to reconstruct how the usage of the information contained in the original images has evolved in time and space, thus permitting to identify, for instance, how these images have been used by groups of people with different opinions about the original event. We start by considering an event occurring in the real world, the *true scene*; such an event could be temporally and spatially extended but here we focus on an event occurring at a fixed time and seen from a particular viewpoint. Let us suppose that a set of images representing the specific *true scene* is available. We are interested in finding dependencies among such images in order to construct a sort of graph which could help to know image histories and relationships.

But what "finding dependencies" means? We would like to understand if one image comes from the other and the processing which produced such a transformation, i.e. if there is some relationship between the two images. However, since the images represent the same content, it is important to highlight that the relationship we are looking for should not be related to the content itself; otherwise, all the images representing the same scene would be judged as dependent.

In order to better explain the rationale behind our approach, let us give an example. Let us consider two artists working on two different paintings; if they are free to paint any possible subject, then the possibility that the two painters draw the same topic is extremely low, and could be taken as an evidence that some form of communication (or some dependency) between the painters occurred. On the contrary, if the subject of the painting has been imposed to the artists, then the same content can not be considered as a demonstration that they talked to each other or that one of them copied the work of the other. But if they painted the same content by using exactly the same colors and the same pictorial metaphors, then we could conclude that the painters had some kind of contact or that one artist copied the other. Here we are interested in the second case, that is, we look for a form of dependency not depending on image semantic.

To do so, we will suppose that any image can be described as the composition of two contributions, namely a component conveying the content information related to the *true scene* and a content-independent component representing some peculiar characteristics of the way the images have been produced, then finding the dependency between two images representing the same true scene means, for our purpose, measuring the similarity between their content-independent components. In the following we propose a rigorous formalization of the above concept.

3. PROBLEM FORMALIZATION

We consider a set of images $\mathcal{I}$, where an image $I \in \mathcal{I}$ is a $N \times M$ matrix (N and M finite and bounded values $\in \mathbb{N}$), whose entries are integer values $\in [0, 255]$. We consider a set of fundamental image processing functions (f-IPFs) Φ_f, consisting of a number of functions ϕ_f, described as: $\phi_f(\cdot) : \mathcal{I} \times \wp_{\phi_f} \rightarrow \mathcal{I}$, where $\wp_{\phi_f}$ is the set of parameters characterizing the f-IPF. The domain $\mathcal{D}_{\phi_f}$ of ϕ_f is the set of input images on which the f-IPF can work, the codomain $\mathcal{C}_{\phi_f}$ is the set of output images defined as: $\mathcal{C}_{\phi_f} : \{I \in \mathcal{I} : \exists I^* \in \mathcal{I}, \exists \wp^*_{\phi_f} : I = \phi_f(I^*, \wp^*_{\phi_f})\}$.

Given two f-IPFs $\phi_1 : X \rightarrow Y$ and $\phi_2 : V \rightarrow Z$, they can be composed by firstly applying ϕ_1 to an argument x and then applying ϕ_2 to the result: $\phi_2(\phi_1(x))$, achieving a composite image processing function (c-IPF) $\phi_3 = \phi_2 \circ \phi_1 : X \rightarrow Z$.

Note that the codomain of ϕ_1 must be included in the domain of ϕ_2: $Y \subseteq V$. We define the set of composite image processing functions (c-IPFs) Φ_c as a set of all the possible compositions of a number r of f-IPFs $\in \Phi_f$, where r is the maximum order of the composition. From now on we consider the simplified case $r = 2$ and we are interested in the set Φ that includes a set of f-IPFs $\in \Phi_f$ and/or a set of c-IPFs $\in \Phi_c$.

Finally, given an image $I_k \in \mathcal{I}$ and an image processing function $\phi \in \Phi$, we say that I_k is compatible with ϕ if $I_k \in \mathcal{C}_\phi$.

3.1. Dependency test

The main idea on which we base our formalization is the following. Let us consider a shot of a real event, that we call *true scene* and refer as $\mathcal{T}$. Let us consider a set of images $\mathcal{I}$ that are the representation of $\mathcal{T}$. Our interest is to find if such images are pairwise dependent.

As we anticipated, we make the hypothesis that any image I belonging to $\mathcal{I}$ can be univocally described as the composition of two separable and independent parts: $[I]_\mathcal{C}$ describing the content of the *true scene* and $[I]_\mathcal{R}$ representing the content-independent characteristics of the image, a sort of random part of the image:

$$I \leftrightarrow \big[[I]_\mathcal{C}, [I]_\mathcal{R}\big] \quad \forall I \in \mathcal{I}. \tag{1}$$

To verify the dependency between two images I_A and I_B, hereafter considered as two random information sources, we consider their mutual information:

$$\mathrm{I}(I_A; I_B) = H(I_A) - H(I_A|I_B) \tag{2}$$

where $H(I_A)$ is the entropy of the source I_A and $H(I_A|I_B)$ is the conditional entropy of the source I_A conditioned to I_B.

By representing the images through the independent components introduced before, equation (2) can be rewritten as:

$$\mathrm{I}\big(\big[[I_A]_\mathcal{C}, [I_A]_\mathcal{R}\big]; \big[[I_B]_\mathcal{C}, [I_B]_\mathcal{R}\big]\big) = \tag{3}$$

$$= H\big(\big[[I_A]_\mathcal{C}, [I_A]_\mathcal{R}\big]\big) - H\big(\big[[I_A]_\mathcal{C}, [I_A]_\mathcal{R}\big] \mid \big[[I_B]_\mathcal{C}, [I_B]_\mathcal{R}\big]\big) \tag{4}$$

$$= H([I_A]_\mathcal{C}) + H([I_A]_\mathcal{R}) - H\big([I_A]_\mathcal{C} \mid [I_B]_\mathcal{C}, [I_B]_\mathcal{R}\big) - H\big([I_A]_\mathcal{R} \mid [I_B]_\mathcal{C}, [I_B]_\mathcal{R}, [I_A]_\mathcal{C}\big) \tag{5}$$

$$= H\big([I_A]_\mathcal{C}\big) + H\big([I_A]_\mathcal{R}\big) - H\big([I_A]_\mathcal{C} \mid [I_B]_\mathcal{C}\big) - H\big([I_A]_\mathcal{R} \mid [I_B]_\mathcal{R}\big) \tag{6}$$

$$= \mathrm{I}\big([I_A]_\mathcal{C}; [I_B]_\mathcal{C}\big) + \mathrm{I}\big([I_A]_\mathcal{R}; [I_B]_\mathcal{R}\big) \tag{7}$$

where (5) has been achieved by exploiting the independence between $[I_A]_\mathcal{C}$ and $[I_A]_\mathcal{R}$ and the chain rule [7] and (6) has been achieved by exploiting the independence between $[I_A]_\mathcal{C}$ and $[I_B]_\mathcal{R}$ and the independence between $[I_A]_\mathcal{R}$ and $[I_A]_\mathcal{C}$ and $[I_B]_\mathcal{C}$.

The mutual information between the images can thus be expressed as the sum of the mutual information between the content components and the random components. For our analysis, we only consider the second term (the content-independent one), since the first term will never be null, due to the inherent dependency between the $\mathcal{C}$ components (since they refer to the same true scene).

We can now cast the problem of finding the dependency between I_A and I_B as a hypothesis testing problem, in which we want to test the hypothesis that $[I_A]_\mathcal{R}$ and $[I_B]_\mathcal{R}$ are independent, that is we want to test whether $\mathrm{I}([I_A]_\mathcal{R}; [I_B]_\mathcal{R}) = 0$. The design of an optimal criterion for such a test would require the existence of a good model to describe $[I_A]_\mathcal{R}$ and $[I_B]_\mathcal{R}$ and their possible relationship. This is a very complicated task, hence we simplify the analysis by adopting as sufficient statistic the correlation ρ between $[I_A]_\mathcal{R}$ and $[I_B]_\mathcal{R}$.

More precisely, by considering a set of image processing functions Φ, we make the assumption that if there is some relationship between two images $I_A, I_B \in \mathcal{I}$, then one image, or a part of it, is given as the processing of the other by means of a $\phi_j \in \Phi$. By relying on such hypothesis, the dependency of I_A on I_B is evaluated by comparing I_A with all the possible processing of I_B through the functions in Φ, namely $\phi_j(I_B)$. In particular, we compute the correlation ρ_j between $[I_A]_\mathcal{R}$ and $[\phi_j(I_B)]_\mathcal{R}$, for each $\phi_j \in \Phi$, and use as sufficient statistic $\mathcal{D}$ the maximum correlation value in this set of correlations:

$$\mathcal{D}(I_A; I_B) = \max_{\phi_j \in \Phi} \rho([I_A]_\mathcal{R}, [\phi_j(I_B)]_\mathcal{R}) \tag{8}$$

where the parameters $\wp_{\phi_j}$ characterizing ϕ_j are here omitted for simplicity. Let us note that the previous statistic is voluntarily asymmetric, i.e. it tests the dependency of I_A on I_B and not the reverse.

To accept/reject the hypothesis of independence we compare $\mathcal{D}$ with a suitable threshold $\mathcal{T}$ and the relative *dependency test* is:

$$\begin{cases} \mathcal{D}(I_A; I_B) < \mathcal{T} & I_A \text{ is independent of } I_B \\ \mathcal{D}(I_A; I_B) > \mathcal{T} & I_A \text{ is dependent on } I_B \end{cases} \tag{9}$$

Actually, we estimate the dependency among images by measuring the *correlation coefficient* and by considering its absolute value: thus, the corresponding sufficient statistic results in a normalized value $\mathcal{D}_N(I_A; I_B) \in [0, 1]$ that replaces equation (8). The relative threshold $\mathcal{T}_N$ to be used in the *dependency test* could be fixed rigorously by studying the statistical characteristics of $\mathcal{D}_N$ and by fixing a value for the false positive probability.

3.2. Dependency graph

Given a set of images of the same scene, we would like to build a graph for representing the relations between them. In the most common sense of the term, a graph is an ordered pair $G := (V, E)$ comprising a set V of vertices or nodes together with a set E of edges or lines, which are 2-element subsets of V. Graphs are simply a collection of nodes and edges connecting the nodes. In our case, the set of nodes V represents the images, the edges represent the dependency relationships between them. Edges can be undirected, meaning the relationship between images holds both ways, or they can be directed, meaning the relationship is one way. The graph is weighted if a weight is assigned to each edge. In our problem, each edge can be labelled with the corresponding sufficient statistic value and with the image processing function relating the corresponding images. The graph, as data structure as well as visual representation for the image dependency relationship seems a natural choice: by using a graph we inherit all the existing algorithms for querying the history, in terms of relations, of a picture in a specific context. By giving the graph a semantic nature/setting we could also build a set of rules (an *ontology*) to infer other relations between images. If we design the graph according to the so called semantic web principles, its links and ontology could be shared to other web applications.

3.3. The graph construction

For each considered set of image-nodes ($\mathcal{I}$), once established the set Φ and an appropriate threshold $\mathcal{T}_N$, the dependency graph is set up from the collection of dependency test results on them.

An exhaustive and straightforward approach to collect such results, is to evaluate the dependency between all the image pairs by considering all the processing functions in Φ, in order to find the maximum correlation value as detailed in Sec. 3.1. To considerably improve the performances, the selection of an adaptive set Φ for the test between couple of images is crucial; we here propose to select for the evaluation of the dependency of I_A on I_B, a subset $\Theta_{I_A} \subseteq \Phi$ formed only by those functions I_A is compatible with (as defined in Sec. 3) by means of an image forensics driven strategy.

The image forensic algorithms proposed so far, through the analysis of a single image $I_k \in \mathcal{I}$, are able to obtain: i) the source σ_r that generated I_k, where $\sigma_r \in \Sigma$ is the set of possible digital image sources (e.g.digital camera, cell-phone, scanner, computer graphic); ii) a subset of image processing functions $\Theta \subseteq \Phi$ a given $I_k \in \mathcal{I}$ is compatible with. It is possible to represent schematically a forensic algorithm by means of a block, that we call Forensic Unary Block (FUB), characterized by INPUT: $I_k \in \mathcal{I}$ and OUTPUT: I_k is/is not compatible with σ_r/θ_r with parameters $\wp_{\sigma_r/\theta_r}$, for each $\sigma_r \in \Sigma$ or $\theta_r \in \Theta$. By exploiting a set of available FUBs, we could apply a procedure that first inspects all the images $\in \mathcal{I}$ by means of all the FUBs in order to achieve information about compatibility between images and functions, and then to evaluate the dependencies by testing only compatible image processing functions.

Such an approach reduces the system complexity by allowing to check only the compatible processing functions to assess the dependency of I_A on I_B. It worths to be noted that a further advantage comes from such a strategy adoption: the approach allows to exclude (a priori) wrong dependency links, indeed it could exist a $\phi \in \{\Phi \backslash \Theta_{I_A}\}$ that gives a high value of $\mathcal{D}_N(I_A; I_B)$.

After collecting the dependency test values for each pair of images in $\mathcal{I}$ (either applying the complete set Φ of functions or the subset Θ), a first version of the dependency graph is built, by taking only those links whose dependency is over the threshold $\mathcal{T}_N$.

By critically observing the relationships, it is possible to modify the graph by reasoning on it, by noticing logical edges that the system fails to find or solve automatically; thus, a rule based modification of such graph could result in a more correct representation of the images history. An *ontology* could allow the system to automatically infer higher order relations between images and to solve ambiguous image relations.

4. A BASIC EXAMPLE

To demonstrate how the proposed formalization could be implemented in practice, we consider now a simple practical example, where a limited set of images $\mathcal{I}$ represents a *true scene*. By applying the proposed dependency test we are able to build the graph where the links indicate the dependency among the images in $\mathcal{I}$. By focusing on a restricted set Φ of processing functions, we test the dependencies among all the possible pairs of images in $\mathcal{I}$ by applying an exhaustive approach on Φ, not the best choice as explained.

4.1. Experimental setting

In our simple example, the set $\mathcal{I}$ consists of six images (see Fig. 1): I_A and I_B are two independent natural images (taken by a digital camera with its native JPEG compression not considered as a processing), while the remaining four are computed by post processing I_A and I_B through the following functions by using Matlab® (see the graph in Fig. 2):

- I_C is obtained by JPEG compressing I_A with a 54% quality factor (Q);
- I_D is obtained by first applying a contrast enhancement function to I_B then JPEG compressing it with Q=90%;
- I_E is the composition of the bottom half part of I_A and the top half part of I_D saved with a JPEG compression with Q=72%;

- I_F is obtained by first applying a 2 degrees clockwise rotation to I_C then JPEG compressing it with Q=98%.

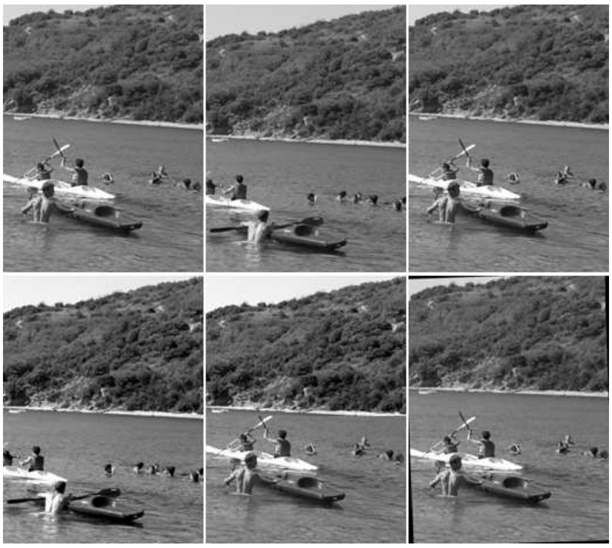

Fig. 1. Example of $\mathcal{I}$ consisting in six images representing the *true scene* ($I_A \ldots I_F$ left $\rightarrow$ right, top $\rightarrow$ bottom).

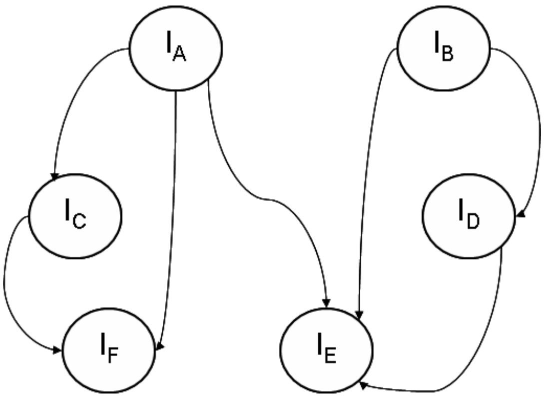

Fig. 2. The original graph representing dependencies among the six images $\in \mathcal{I}$.

To make the experiment more realistic we suppose that each processing to any image is always followed by a JPEG compression: thus, when we consider a composite image processing function in Φ, the last operation is always a compression. Furthermore, for sake of simplicity we consider as distinct processing functions the same process with different parameter values, e.g. a JPEG with Q=50% and one with Q=90% are considered as two distinct $\phi \in \Phi$. The choice of the set Φ considerably influences the results of the dependency test (and also the system complexity); we consider a set of image processing functions that includes good approximations of the processing actually used for achieving $\mathcal{I}$, namely:

- $\phi_1 \ldots \phi_6$: JPEG compression with Q $\in [50\%, 60\%, 70\%, 80\%, 90\%, 100\%]$;
- $\phi_7 \ldots \phi_{12}$: contrast enhancement followed by JPEG compression with Q $\in [50\%, 60\%, 70\%, 80\%, 90\%, 100\%]$;
- $\phi_{13} \ldots \phi_{24}$: rotation of $[-2, -1, +1, +2]$ degrees followed by a JPEG compression with Q $\in [60\%, 80\%, 100\%]$;
- ϕ_{25}: the identity function.

Our analysis is based on the assumption that any I can be univocally described as the composition of two independent parts describing the content and the randomness of the image; such an hypothesis is obviously ideal and in practice we must search for a good method that approximatively decomposes any image I into the components $[I]_\mathcal{C}$ and $[I]_\mathcal{R}$. The approach we adopted in our experiment is to apply a Wiener filter to I and let $[I]_\mathcal{R} = I - \textit{Wiener}(I)$.

4.2. Results and discussion

In Table 1 the sufficient statistic $\mathcal{D}_N$ between each couple of images in $\mathcal{I}$ is shown, by computing the random component using the Wiener filter as implemented by Matlab® (that uses a pixel-wise adaptive Wiener method based on statistics estimated from a local neighborhood, here a 5×5 window). The ϕ that produced the maximum correlation coefficient is also shown in the table. Furthermore, in Fig. 3 the correlation coefficient values for all the $\phi \in \Phi$ are plotted for three couple of images. It is possible to note that similar processing functions exhibit similar output values: even if the actual function producing the analyzed image is not included in Φ, the dependency test gives a positive result since the functions in Φ are sufficiently close to the ones actually used to make the test set.

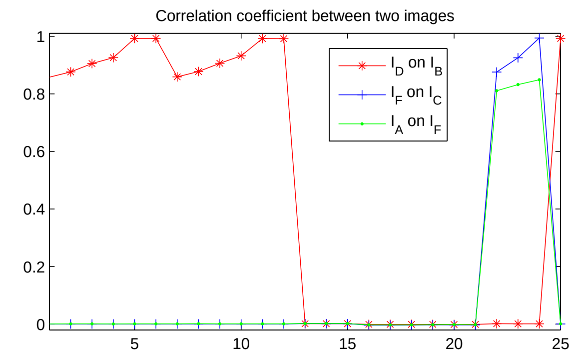

Fig. 3. The correlation coefficients for all the $\phi \in \Phi$ are plotted for the dependency of 3 image pairs: I_D on I_B, I_F on I_C, I_A on I_F.

In order to build the dependency graph we must fix the detection threshold: without resorting to a theoretical approach,

	I_A	I_B	I_C	I_D	I_E	I_F
I_A	1.000, ϕ_{25}	0.004, ϕ_{12}	0.957, ϕ_1	0.004, ϕ_{11}	0.456, ϕ_3	0.849, ϕ_{23}
I_B	0.004, ϕ_{25}	1.000, ϕ_{25}	0.003, ϕ_7	1.000, ϕ_{11}	0.517, ϕ_9	0.003, ϕ_{24}
I_C	0.852, ϕ_{11}	0.003, ϕ_2	1.000, ϕ_{25}	0.003, ϕ_2	0.404, ϕ_7	0.994, ϕ_{24}
I_D	0.004, ϕ_{25}	0.993, ϕ_{25}	0.003, ϕ_1	1.000, ϕ_{25}	0.524, ϕ_3	0.003, ϕ_{23}
I_E	0.429, ϕ_{25}	0.482, ϕ_{25}	0.409, ϕ_4	0.484, ϕ_{25}	1.000, ϕ_{25}	0.403, ϕ_{24}
I_F	0.840, ϕ_{14}	0.003, ϕ_{13}	0.978, ϕ_{14}	0.003, ϕ_{13}	0.3996, ϕ_{14}	1.000, ϕ_{25}

Table 1. Values of $(\mathcal{D}_N, \phi)$ between each couple of images in $\mathcal{I}$: dependency is evaluated by comparing images in the column with all the possible processing of images in the row.

we now empirically chose its value; in particular, by letting $\mathcal{T}_N = 0.35$, we obtained the graph in Fig. 4, where for the two edges referring to the same pair of images, we used solid and dashed lines for larger and lower values respectively. Comparable results have been achieved by changing both the Wiener window size (namely from 5 to 3), and the two starting natural images (namely I_A and I_B).

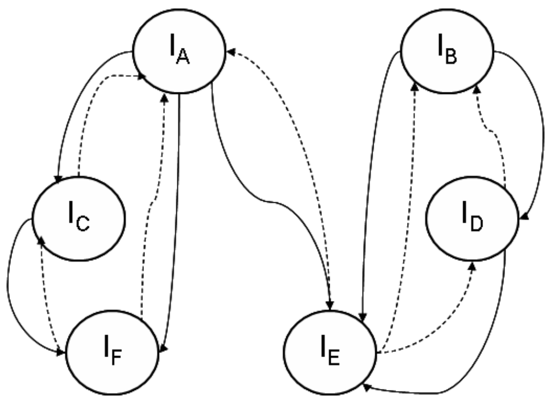

Fig. 4. The obtained graph basing on the results shown in Table 1 and imposing $\mathcal{T}_N = 0.35$.

Note that low values of $\mathcal{D}_N$ could reveal that the corresponding image processing function has been applied to only a subpart of one image for producing the other, thus globally decreasing the correlation between them (as in the case of I_E).

By comparing the original graph in Fig. 2 with that achieved experimentally, we note the numerous relationships present in Fig. 4. We must observe that some second order relationships between the images could exist even though not directly represented in the original graph, e.g. there is a hidden relation between I_B and I_E ($I_B \rightarrow I_D \rightarrow I_E$). The presence of many links between the images also depends on the exhaustive research among all the image processing functions in Φ. As explained in Section 3.3, by using the FUB strategy and by considering only the subset Θ, some wrong edges could be excluded a priori. For example, let us consider the dependency analysis of I_A on I_F: by applying a set of FUBs attesting if the image is compatible with resampling and JPEG compression, since I_A is a non processed natural image, it would result to be non-compatible with the resampling operation. In Fig. 3 the plot of I_A on I_F is above the threshold only for $\phi_{22}, \phi_{23}, \phi_{24}$, but they are not in Θ_{I_A} after the FUBs application. According to this approach, we obtain a graph without the edges from I_F to I_A and from I_F to I_C.

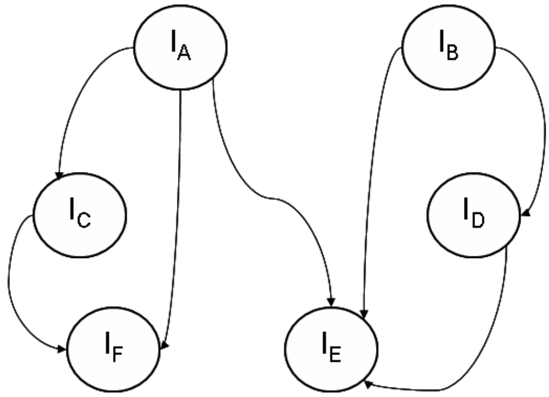

Fig. 5. The simplified graph after the application of the proposed strategy.

The number of edges can be afterwards modified using an *ontological reasoner* implementing a given set of rules: in particular, by comparing the correlation coefficients of the edges referring to the same image pairs, we could remove the edge with the lower value (the ones shown with dashed lines in Fig. 4). The final graph is the one shown in Fig. 5 where the only differences with respect to the original graph are the edges from I_A to I_F and from I_B to I_E, the second order relationships.

5. CONCLUSIONS

We have proposed a roadmap for research in finding dependencies between images: to the best of our knowledge it is the first time that the similarity between content independent components of images has been handled. The proposed approach, relying on several simplifying hypotheses, can be seen as a modular system, composed by some key building blocks: the decomposition method for achieving $[I]_{\mathcal{C}}$ and $[I]_{\mathcal{R}}$; the adopted sufficient statistics; the strategy for the choice of the set of image processing functions used in the dependency test; the ontology applied for the graph analysis. Future work will be devoted to a deeper analysis of these blocks. Moreover, an interesting approach to be examined could be to apply the dependency test block wise, to improve the analysis especially in case of image splicing.

6. REFERENCES

[1] D. Kundur and D. Hatzinakos, "Digital watermarking for telltale tamper proofing and authentication," *Proceedings of the IEEE*, vol. 87, no. 7, pp. 1167–1180, July 1999.

[2] M. Barni and F. Bartolini, *Watermarking Systems Engineering: Enabling Digital Assets Security and other Applications.*, Marcel Dekker, 2004.

[3] P. Blythe and J. Fridrich, "Secure digital camera," in *Digital Forensic Research Workshop*, Baltimore, MD, 2004.

[4] A. C. Popescu and H. Farid, "Statistical tools for digital forensic," in *Proc. 6th Int. Work. on Information Hiding, IH'04*, Toronto, Canada, May 2004, vol. 3200, pp. 128–147, Springer-Verlag.

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

[6] A. Swaminathan, M. Wu, and K.J.R. Liu, "Digital image forensics via intrinsic fingerprints," *IEEE Trans. on Information Forensics and Security*, vol. 3, no. 1, pp. 101–117, January 2008.

[7] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, John Wiley & Sons, New Jersey, Second Ed., 2006.