Pattern Recognition 74 (2018) 326–339

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/patcog

A knowledge based architecture for the virtual restoration of ancient photos

E. Ardizzone, H. Dindo, G. Mazzola*

Dipartimento di Innovazione Industriale e Digitale - Ingegneria Chimica, Gestionale, Informatica, Meccanica (DIID), Università degli Studi di Palermo, Viale delle Scienze building 6, 90128, Palermo, Italy

ARTICLE INFO

Article history:
Received 29 July 2016
Revised 8 September 2017
Accepted 19 September 2017
Available online 21 September 2017

Keywords:
Image restoration
Historical photos
Digitization
Ontology
Knowledge base

ABSTRACT

Historical images are essential documents of the recent past. Nevertheless, time and bad preservation corrupt their physical supports. Digitization can be the solution to extend their "lives", and digital techniques can be used to recover lost information. This task is often difficult and time-consuming, if commercial restoration tools are used for the purpose. A new solution is proposed to help non-expert users in restoring their damaged photos. First, we defined a dual taxonomy for the defects in printed and digitized photos. We represented our restoration domain with an ontology and we created some rules to suggest actions to perform in case of some specific events. Classes and properties of the ontology are included into a knowledge base, that grows dynamically with its use. A prototypal tool and a web application version have been implemented as an interface to the database, and to support non-expert users in the restoration process.

1. Introduction

The art of photography is more than 150 years old, but it absorbed quickly the technological innovations of the following years. Historical photos constitutes a significant part of our Cultural Heritage, as well museums, paintings, archaeological sites. Time and careless preservation methods corrupt physical supports, therefore solutions must be found in order to protect their economic worth and high cultural value. Digitization is the definitive solution to preserve historical images and their content. Digital copies last almost forever, since they can be used and duplicated without losing quality. Furthermore, digital restoration techniques can be used to take images back to their original state. Professional digital restorers often use commercial software tools, like Adobe Photoshop, but they are neither automatic nor user friendly. In fact defects are subjectively detected, and correction methods are selected by user as well. This task is complex, expensive, and acceptable just for very important pictures.

In literature some general approaches that dealt with the problem of the virtual artworks restoration have been proposed. Landon et al. [1] presented a system to acquire and restore deteriorated film negatives, focusing onto the digitization process. Del Mastio et al. [2] developed a tool for processing cultural heritage images, with the goal to protect the digital reproductions of artworks, by means of a watermarking technique, and for the virtual restoration of some typical defects. Cislariu et al. [3] developed a tool for the detection and the restoration of digitized wood paintings. Marionidis et al. [4] presented an integrated tool for damage detection, shape restoration and texture restoration of faces appearing in Byzantine icons. Samko et al. [5] proposed a system to process fragile scrolled historical documents, without the need to physically unravel them, and to recover partially lost information. Hedjam and Cheriet [6] proposed a new approach to acquire images from historical documents by using a multispectral imaging system, and then restore them combining information from the different channels. The previous approach by Moghaddam and Cheriet [7] was based on an adaptive multi-scale binarization algorithm. Bhardwaj et al. [8] presented a new technique for photo defect detection, by sensing the light after it passes through the photo and then thresholding the digitized image to identify defect objects. Zhang et al. [9] proposed a framework, based on an inpainting algorithm, and a shape-from-shading technique, to remove unwanted distortions in digitized image from printed documents. Some state of the art algorithms for the detection and the restoration of specific defects in digitized degraded photos will be discussed in Section 2.

The adopted approach consists in the definition of an ontology to represent objects that are involved in restoration process of digitized degraded old photos. Ontology engineering has been pre-

* Corresponding author.
E-mail addresses: edoardo.ardizzone@unipa.it (E. Ardizzone), haris.dindo@unipa.it (H. Dindo), giuseppe.mazzola@unipa.it, giuseppe.mazzola75@gmail.com (G. Mazzola).

https://doi.org/10.1016/j.patcog.2017.09.031
0031-3203/

viously applied to the image processing, for example, domain in [10], in which a photo annotation ontology is used to help users in querying images, and in [11], for the art image retrieval problem.

Photo restoration can be made automatic by finding a way of representing knowledge in order to formalize the pair "kind of defect-recovering algorithm". Hence, it is necessary to design a scheme which enables to select the proper restoration methodology starting from the interpretation of the defect meta-representation. Such a scheme constitutes an ontology in which the need of restoration, expressed by the image meta-representation, is satisfied by means of experience coming from knowledge. Starting from the ontological representation, a knowledge-based model is created to guide non expert users through the restoration process of the degraded artworks. In particular we analysed photos from Alinari Archives in Florence, which is composed of high resolution, colored or black and white images since 1840.

The proposed approach presents some completely novel tools for digital image restoration:

- while manual restoration process of printed photos is based on the physical-chemical causes of defects, digital restoration algorithms process digital damage features. Our approach let us to correlate damages in printed and digitized photos;
- the restoration process is based on knowledge and then it is able to receive experience about new kinds of defects along with more recent and effective algorithms for restoration: the model constantly grows while it is used;
- the meta-representation of image damages can be also used in finer applications, as content retrieval or the automatic definition of pictures degradation typologies;
- a basis of knowledge is defined to automatically select the best restoration method for an image which is affected by a specific damage.

2. Damage taxonomy

Old images may present a huge variety of damages, due to several different factors. Some defects may lead to a complete loss of information, while others deteriorate the overall appearance of images. Mostly, damages are originated by inappropriate environmental conditions (temperature, humidity, lighting), inaccurate handling (dirt, image protection, cracks), human intervention (stamps, writings, restorations) and chemical factors (reactions with micro-organisms). While the origin of image defects on the physical support (whether positive or negative) is an important issue for a manual restoration activity, several defects appear similar once images are digitally scanned (Fig. 1) and can be described and removed by similar underlying processes. Several works studied specific defects of digitized/digital pictures: cracks and craquelures [12–14], water blotches and foxing [15], semi-transparent blotches [16], fading [17,18], scratches in photos and movies [19,20], ink sprays and scratches [21], color [22,23]).

A first interesting attempt to classify defects in old photos was proposed in [24], but it was incomplete and it did not focus on the digital aspect of the defects. In [25] we proposed our origin-based defect taxonomy, but it was not either based on digital features (shape, color, texture, etc.). Finally, an observer will not be able to discriminate an abrasion from a tear, if their digital versions have similar aspect (see Fig. 1).

Fig. 2 shows how manual restorers annotate their printed documents: each type of defect is identified by a standard numerical code, and each defect is removed using the most appropriate approach. The need of a digital damage taxonomy arises from the aim to associate damages to the most appropriate "digital" detection and restoration algorithms. To analyze the damages of a digital image, instead of studying their origin, digital features (shape, color, texture, etc.) must be analyzed. An automatic inspection method won't be able to discriminate an abrasion from a tear, if their digital versions have similar features (see Fig. 1). Then we proposed [26] two different taxonomies, to distinguish defects in printed photos and in their digital versions, and to compare features of the two defect sets.

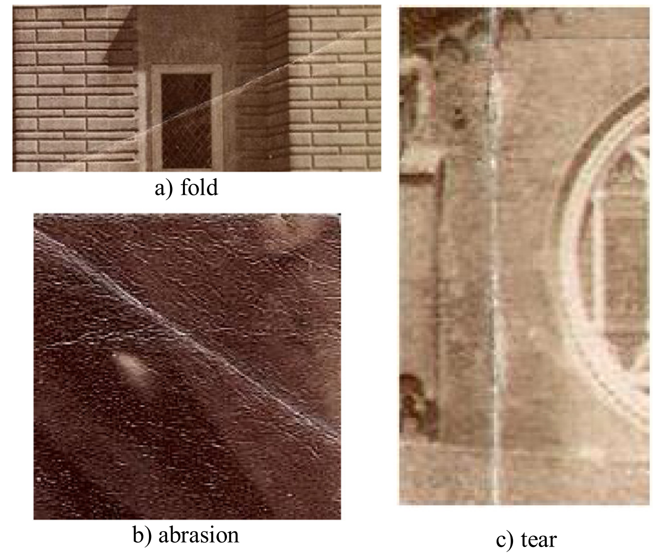


Fig. 1. A comparison between three defects, which have similar digital aspects. In the printed version, according to the physical-chemical origin, they are classified as three different types of defect.

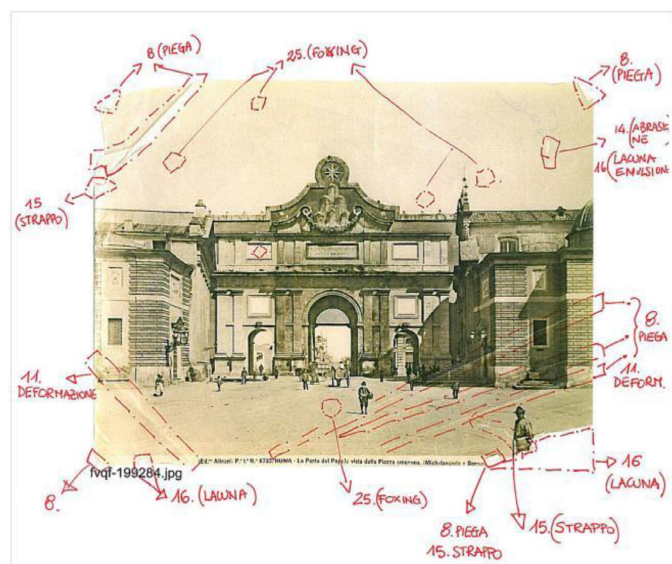


Fig. 2. An example of an annotation file, written (in Italian) by a manual restorer from the Fratelli Alinari Museum Collection (courtesy).

2.1. Real defect taxonomy

Table 1 shows the taxonomy used by expert manual restorers of the Fratelli Alinari Museum Collections in Florence to annotate their photos. Each type of defect is labeled by a standard numerical code, which univocally identifies it (see also Fig. 2). According to their origin, real defects of old photographic prints can be divided in different sets:

Table 1
Real defect taxonomy.

Biological alterations		14	Abrasions
1	Infections	15	Tears
2	Infestations	16	Lacunas
3	Other	17	Cracks
Physical alterations		18	Pr. of adhesive
4	Garbage	19	Other
5	Dust	**Chemical alterations**	
6	Fingerprints	20	Spots
7	Stains	21	Fading
8	Folds	22	Yellowing
9	Craquelures	23	Silver mirroring
10	Lifting	24	Sulfuration
11	Deformations	25	Foxing
12	Bending	26	Other
13	Marks		

2.1.1. Mechanical (physical) damages

Usually originated by inaccurate handling and/or store of the original image, can be further divided into:

- **Deposited matter**: different materials adhere to the surface creating small spots that cover the original image. Some examples are:
- *Dirt & dust*: may be hygroscopic, absorbing water or humidity, causing molds, hydrolysis and acidification;
- *Fingerprints*: the finger impression modifies the emulsion structure; typical shape;
- **Stains**: spots originated by water or humidity; they alter the color and the physical-chemical structure of the phototype components; the presence of water may swell the surface of the support;
- **Physical alteration of images**: often lead to a complete loss of information and should be removed by specialized techniques. Typical examples are:
- *Cracks*: may be very large; do not exhibit a dominant orientation; however, each crack has its own direction;
- *Folds*: caused by improper human intervention; a bend in a material which may cause a crack or break in the emulsion or support;
- *Craquelures*: micro-fractures of the support of the photo, usually branched; orientation depends on that of the paper fibers of the support;
- *Abrasions*: lack in the emulsion of the support, caused by friction with other part of the photo or with some external tool (i.e. scratches);
- *Tears*: caused by improper human intervention; a rip of the support, with or without paper loss;
- **Deformations**: originated by an inappropriate conservation of original images; often caused by excessive humidity and/or temperature and corrupt the way the gelatin is fixed to the support; the effect is a deformation of the planarity of the support
- *Lifting*: the local or diffuse partial detachment of the structural components of a photo from the support;
- *Bending*: curving of the support of the photo;
- **Human retouches**: deliberate human retouches that irremediably alter the image; some examples:
- *Lacunas*: a portion of the photo, or of one of the components, that is totally lost;
- *Marks*: impression that modifies the surface or the structure of the phototype components;
- *Presence of adhesive*: caused by scotch tapes or glue, which degradation alters the color of the support or of the image (to amber-ochre yellow).

2.1.2. Chemical damages

Chemically originated, may be further divided into:

- **Spots**: local or punctual alteration of the original colouring of the image. Examples are:
- *Foxing*: chemical reactions between the print and some microorganisms; appears as reddish-brown spots, with irregular edges, with a darker kernel and a lighter surrounding area;
- *Silver mirroring*: bluish-silver colored spots; more evident in the darkest areas of the image; caused by bad quality of the wrappers or pollution;
- *Sulfuration*: dark-gray blue spots due to chemical reaction between sulphur and metal components of the support; caused by pollution or incomplete washing phase during the photo development;
- **Tonal and color balance**: originated by an excessive exposure of original photos to light; some examples:
- *Fading*: overall whitening of the image; contrast and details of the image are lost;
- *Yellowing*: alteration in the image chromaticity, which tends to yellow; may be also caused by improper handling or preservation.

2.1.3. Biological damages

Deterioration of the support due to the attack of living organism . May be further divided in:

- **Infections**: biological attacks (by fungi, molds or bacteria) to the emulsion or the support; they look like colored spots (yellow, brown, violet, black);
- **Infestations**: losses and staining caused by insects or small rodent feeding on the photographic materials.

2.2. Digital defect taxonomy

Within the proposed digital taxonomy, a damage can be **global**, if it affects every pixel in the image, **diffuse**, if distributed overall the image, without affecting all the pixels, and **local**, if it is located in a limited portion of the image. According to their digital aspect, defects may be classified as:

- **Spots**: local defects. Fig. 3(a). Underlying information is lost and must be fully substituted. No specific colors. More or less rounded shaped.
- **Semi-transparent spots**: local defects. Fig. 3(b). Similar to spots, residual information can be recovered with restoration techniques.
- **Scratches**: local defects. Fig. 3(l). Thin lines, with a preferential direction. Usually lighter than the rest of the image. Can have a darker kernel. Possible limited changing in width and slope.
- **Foxing**: local/diffuse defects. Fig. 3(d). Covering or semi-transparent spots. It is composed by a dark red- brown kernel, surrounded by a lighter red area, which is usually semi-transparent.
- **Folds**: local defects. Fig. 3(g). Located near the edges of the photo. It is composed by a lighter central area (the verso side) and darker edges, depending on the acquisition operation.
- **Cracks**: local defects. Fig. 3(e). Undefined orientation. In some cases they can have branches. Cracks are usually composed by a darker kernel surrounded by a lighter area.
- **Deformations** (lifting): local defects. Fig 3(f). Due to the digital acquisition of a non planar support. They look like the negative of a branched crack.
- **Blotches**: diffuse defects. Fig. 3(c). Semi-transparent spots which can be seen all over the image. Usually lighter than the rest of the image.
- **Whitening**: global defect. Fig. 3(k). Overall fading.
- **Yellowing**: global defect. Fig. 3(j). Distortion in the chromaticity (to yellow) of the whole image.

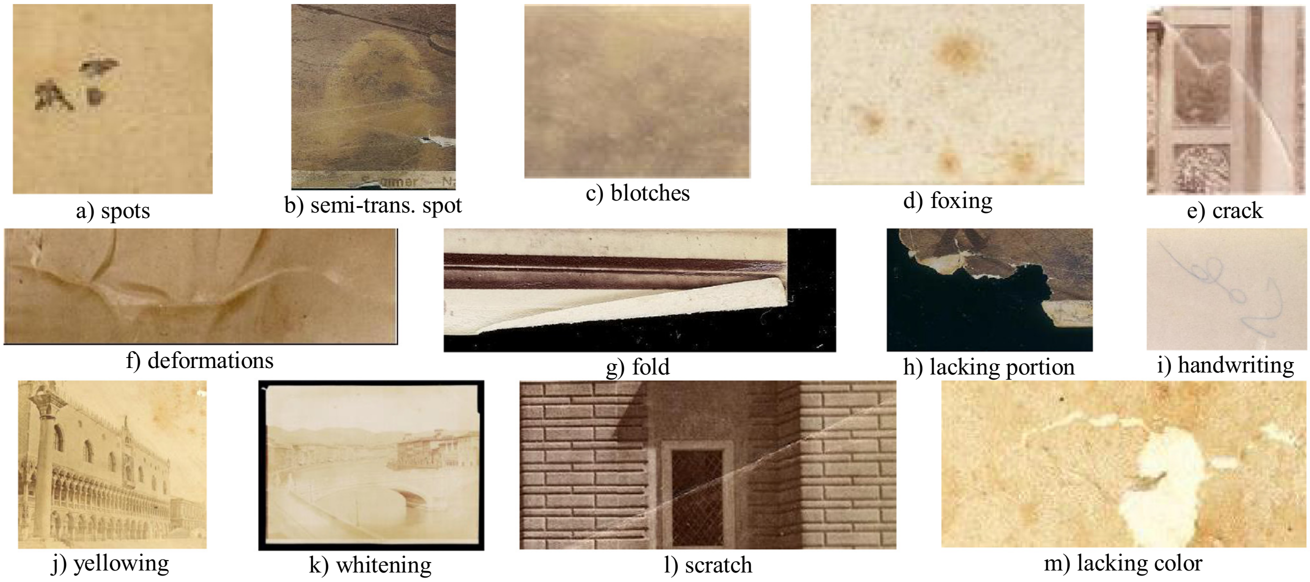


Fig. 3. Some examples of defects classified according to the digital taxonomy (courtesy of Fratelli Alinari Museum Collections (RFMA) in Florence).

Table 2
Digital defect taxonomy. For each digital damage it is shown the percentage of the corresponding real damages, identified by the numerical code of Table 1, in the dataset of printed images.

Digital defects	Type	Real defects
Spots	Local	4(6%), 6(3%), 14(3%), 18(3%), 19(30%), 20(46%), 23(3%), 25(6%)
Semi-transparent spots	Local	20
Scratches	Local	8(10%), 11(15%), 14(50%), 15(15%), 17(10%)
Foxing	Local/Diffuse	25
Folds	Local	14
Cracks	Local	15(75%),11(25%)
Deformations	Local	11
Blotches	Diffuse	7(5%), 20(35%), 23(60%)
Fading	Global	21
Yellowing	Global	22
Lacking color	Local	14(43%),16(57%)
Lacking portions	Local	8(5%), 11(13%), 16(77%), 18(5%)
Handwritings	Local	19

- **Lacking color**: local defect. Fig. 3(m). Undefined shape (some similar to scratches, some other to spots). Usually white, because of the exposition of the color of the support. Information is totally lost.
- **Lacking portions**: local defect. Fig. 3(h). Usually black, but it depends on the acquisition condition. In most cases lacking portions have jaggy edges.
- **Handwritings**: writings and scrawls. Fig. 3(i). Complex curve lines, darker than the rest of the image.

Table 2 lists the digital defects within our testing dataset (about 250 images from the Alinari Photographic Archives in Florence). Table 2 shows also a-posteriori analysis of the relationships between the two proposed taxonomies, we used to annotate our dataset. Note that there is no 1-1 relationship between the defects in the two taxonomies. For example, "digital" spots can be caused by the digital acquisition of defects such as garbage, abrasions or chemical spots. On the other hand, real photos, which had been manually annotated as affected by abrasion, according to the digital defect taxonomy are classified as spots, lacking color, or scratches, according to their digital appearance. It is clear that digital and manual restorers often cannot be in agree about the classification of a defect in an image. Our work aims to make a first attempt to bring closer the digital and manual restorers' points of view.

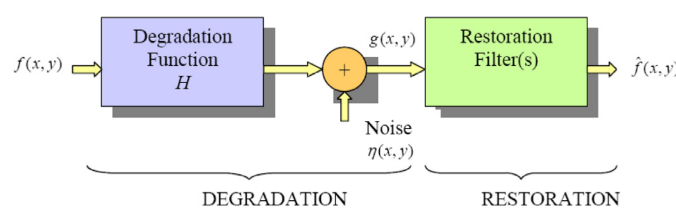


Fig. 4. Classical restoration model.

3. Our restoration model

In the classical approach [27] (see Fig. 4) the degradation process is modeled as a function that, with an additive noise term, operates on the f(x,y) input image to produce the degraded image g(x,y). Given g(x,y), some knowledge about the function H and the noise η(x,y), the goal of the restoration is to have the closest estimation $\hat{f}(x, y)$. This approach shows several problems to be handled. First of all, the degradation function and the additive noise terms are difficult to estimate, so this approach is often inapplicable. Moreover it was designed for typical defects of digital images (blurring, noise, etc) whichare global and diffuse in the whole image. A different approach is needed to handle defects that come from the digitization of printed documents, which are in most case

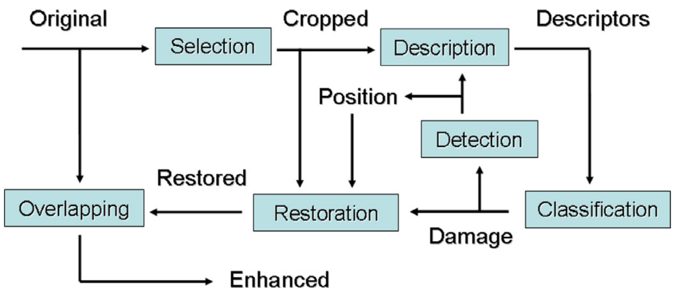


Fig. 5. The proposed restoration model.

local defects, caused by degradation of part of the support of the photo.

To achieve these goals, a new model, which is inspired by the real restoration process of manual photo restorer, is proposed. Fig. 5 shows a simplified scheme of the model:

- **Original image** is the whole input image;
- **Cropped image** is the part of the image which is degraded, and it is selected by the user;
- Cropped image is described with some **descriptors**, and descriptor values are the inputs of the Classification Box;
- **Classification** step aims to recognize by which type of damage the cropped image is affected, using information extracted in the **description** step;
- Appropriate **detection** method is applied, to locate the position of damaged pixels into the cropped image. Detection algorithms are also used to support the description step. Detection is bypassed if a global damage has to be processed;
- Proper **restoration** algorithm is applied;
- The **restored image** is at last overlapped with the original image, to reconstruct the enhanced image.

4. Photo restoration ontology

Ontology is a conceptual representation of a specific domain. It consists of definition of:

- classes (concepts),
- hierarchy and relations between classes,
- properties.

Fig. 6 shows the proposed ontology for the restoration process domain. Protegè Ontology Editor (version 3.4) [28] is used to specify ontology definitions. This ontology has been defined in collaboration with some manual restoration experts from the Alinari Archives in Florence.

4.1. Classes

Four main classes are involved in a restoration process: Image, Damage, Algorithm and Restoration Path:

Image class has 2 subclasses:

- *Printed Image*, which is the primitive photo which had been digitized;
- *Digital Image* can be *Original* (the whole digitized photo), *Cropped* (the selected part to restore), *Restored* (the restored crop), *Enhanced* (the original photo after the restoration of one of its crop or after a global enhancement processing), and the support *Damage Mask* (which is used by local restoration algorithm to find the position of damaged pixels to process);

Damage can be real or digital, which sub-types are described in Section 2;

Algorithm has 5 subclasses:

- *Selection* algorithms are used to choose which degraded portion of the image to restore;
- *Description* algorithms extracts digital features from an image;
- *Detection* algorithms find the position of damaged pixels in an image;
- *Restoration* algorithms are the local correction methods to apply;
- *Enhancement* algorithms are global enhancement methods (contrast enhancement, histogram stretching), which include also the overlapping process of the restored crop into the original image.

RestorationPath is made of algorithms. It is the key class of the restoration process, because it represents a complete method (from detection to restoration) to correct a specific damage, and can be used to restore images which are affected by similar defects.

4.2. Properties

We assigned some properties to the proposed classes, in order to express the relationships that occur between them:

Image properties typically correlate image classes. Examples are:

- isOriginalImageof – isCroppedImageof
- isPrintedImageof – isDigitzedImageof
- isDamagedImageof – isRestoredImageof

The *similarto* property between images is used into the classification step and will be further described in the Section 6. RestoredImage *hasqualityresult* property is used to vote the result of the applied restoration path .

Algorithm properties. Each algorithm class is specified by the input domain to which *isappliedto*(inv. *isprocessedby*) and the output it *returns*(inv. *isreturnedby*):

- Selection algorithm *isappliedto* OriginalImage and *returns* CroppedImage;
- Detection algorithm *isappliedto* CroppedImage and *returns* DamageMaskImage;
- Restoration algorithm *isappliedto* CroppedImage and *returns* RestoredImage, eventually using DamageMaskImage to know the location of damaged pixels to correct;
- Enhancement algorithm *isappliedto* OriginalImage and *returns* EnhancedImage, eventually using a RestoredImage to substitute the damaged crop;
- Description algorithm *isappliedto* CroppedImage and extracts low-level features of the image.
- An Algorithm *isinpath* and *hasorderinpath* of a specific RestorationPath, and eventually *hasinputparameters*.

Damage main property correlates damages and images. Real damages *affects*(inv. *isaffectedby*) PrintedCrop, digital damages *affects* (inv. *isaffectedby*) CroppedImage. DigitalDamage *hasseverity* property is used to give an evaluation of the damage severity. Damages are correlated to DescriptionAlgorithm by the *isdescribedby* property. This property represents the ability of some descriptors to describe in a better way some damages than others. This description ability will be useful in the future for the development of a more sophisticated damage classifier.

RestorationPath *iscomposedby* instances of the Algorithm class, *restored* some CroppedImage and *hasrestorationability* for some type of damages.

5. Rules

Classes and properties represent actors and roles of the analyzed domain. Rules represent actions to make in response to specific events. We use SWRL language [29] to express rules and the

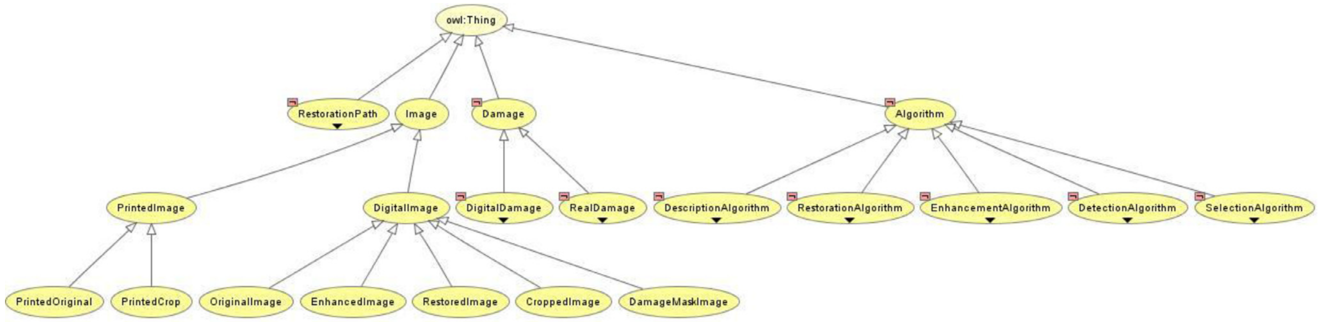


Fig. 6. The ontological scheme of the restoration process (some sub-classes are hidden to improve clarity of the figure). The scheme has been plotted using the Protegè OWLWiz plugin.

D.1: *similarto(?x, ?y) ∧ isaffectedby(?x, ?z) → isaffectedby(?y, ?z)*

R.1: *RestorationPath(?x) ∧ Iscomposedby(?x, ?y) ∧ isappliedto(?y, ?z) → restored(?x, ?z)*
R.2: *RestorationPath(?x) ∧ restored(?x, ?y) ∧ isaffectedby(?y, ?z) → hasrestorationabilityfor(?x, ?z)*
R.3: *RestorationPath(?x) ∧ hasrestorationabilityfor(?x, ?y) ∧ abox:hasClass(?z, ?y) ∧ abox:hasClass(?z, ?a) ∧ differentFrom(?a, ?y) → hasrestorationabilityfor(?x, ?a)*
R.4: *RestorationPath(?x) ∧ restored(?x, ?y) ∧ isaffectedby(?y, ?d) ∧ abox:hasClass(?d, ?e) ∧ isdamagedimageof(?y, ?z) ∧ hasqualityresult(?z, ?a) → sqwrl:select(?e, ?x) ∧ sqwrl:avg(?a) ∧ sqwrl:orderByDescending(?e) ∧ sqwrl:orderByDescending(?a)*

Fig. 7. The most significant rules used within our framework, written in SWRL [29].

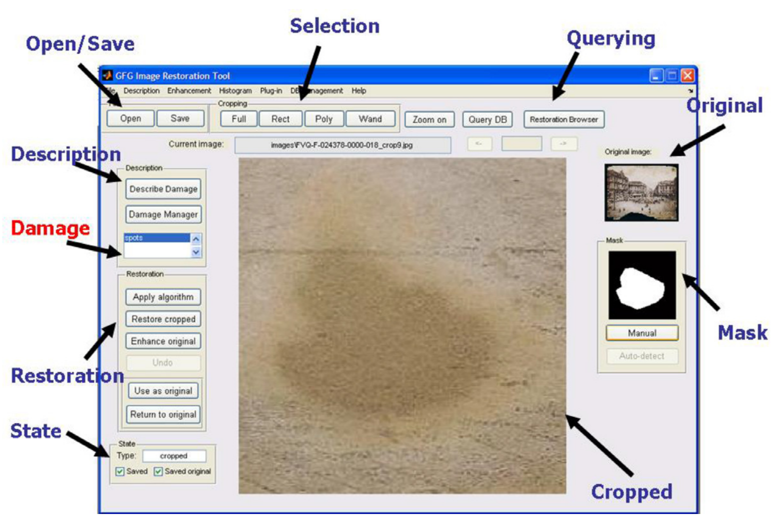


Fig. 8. A screenshot of our prototype restoration tool.

Protegè JessTab Plugin to execute them, using classes and properties of the defined ontology (see Fig. 7). SWRL rules are composed by an antecedent part, which is the activating condition, and a consequent part, which is the action to make when the antecedent fires.

The most significant rules in our framework are:

Damage assignment rule:

- D.1 is used to assign to a new degraded image the damage of the most similar image. The similarity condition will be explained in the next section.

Restoration Path rules:

- R.1 to record the relationship between the applied RestorationPath and the damaged CroppedImage.
- R.2 to assign to a RestorationPath the ability to restore a type of damage that affects the image it processed.
- R.3 to assign to a RestorationPath the ability to restore damages that are similar to those it has already processed.
- R.4 to rank RestorationPath instances which has the ability to restore a specific damage. For a specific damage, RestorationPath instances are sorted according to the average results of the images they restored. These rules will be further discussed in the implementation section.

In the theoretical model, new rules can be easily added to the system using the SWRL language. In practice, each rule is properly implemented in the prototype with a function, then adding a new rule implies writing a new function for the tool.

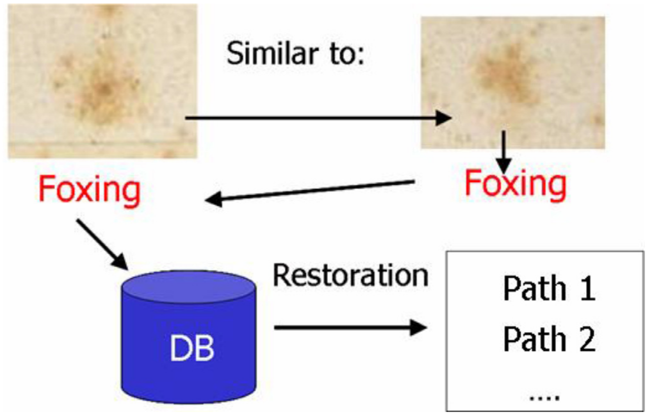


Fig. 9. Steps from classification to restoration path proposal.

6. Prototype implementation

Elements of the proposed ontology are included in a knowledge base, which is recorded in a relational database (for further details see [25]). Entities and relationships corresponds to classes and properties of the proposed ontology. We also developed a user-friendly database management tool (see Fig. 8 for a screenshot) to manage the contents of the database.

Our prototype tool, implemented in Matlab, assists the user in several steps of the restoration process. It is able to describe and identify degradation typologies of an image, to apply several existing algorithms or to add user-defined algorithms for restoration. Many useful queries are implemented and their output can be saved in a custom XML format, or seen as an HTML page. In this section we describe how our framework supports non expert users in a typical restoration process of one of his damaged photo.

A user can load an image (original, cropped, restored, enhanced) from the database, or a new image he wants to restore. If it is a new image, it is saved into the DB, and stored as a new original image. The user selects a part of the image that he evaluates to be corrupted, even if he has no knowledge about the type of damage. The selected part of the image is then saved as a new cropped image. Cropped image can coincide with the original image, in case of global defects.

The next step is to classify the type of damage by which the cropped image is affected. The user may opt to manually assign a damage, and the related severity, using the appropriate plug-in tool, or to use the automatic description tool. In both cases descriptors of the current image are computed, and recorded into the DB. The automatic classification step, in the current version of the tool, is implemented with a Nearest Neighbor classifier. The current image is compared with all the cropped image in the DB, using a weighted Euclidean distance of the descriptors' values. (see [25] for more information about the descriptors we used in our tests). Our system assigns to the current image the damage of the most similar image in the DB, according to this distance (see the top part of Fig. 9), asking for confirmation. We chosen this simple classifier because the knowledge base of our system constantly grows at each use. Then the Nearest Neighbor algorithm is a good solution, in our opinion, to exploit the new information acquired

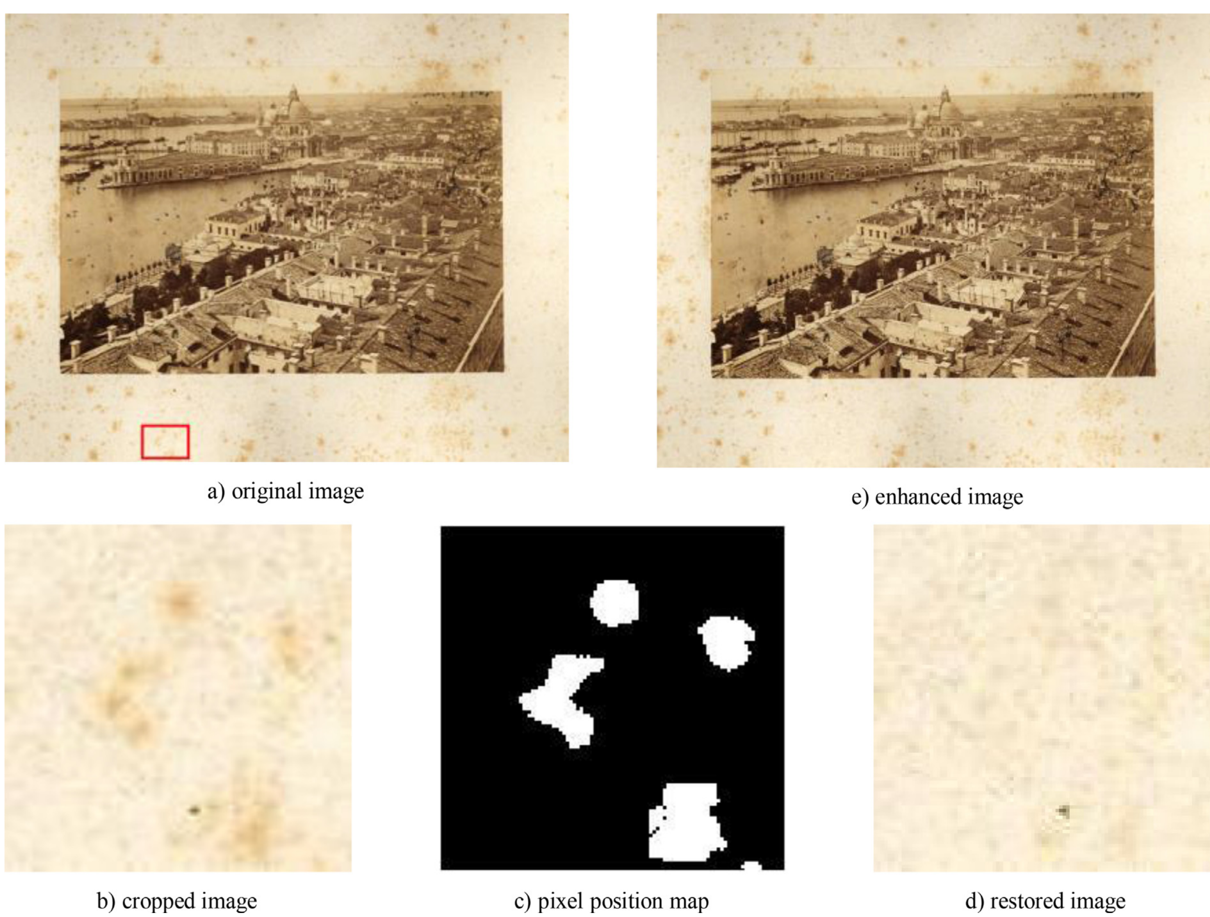


Fig. 10. An example of the restoration process of a damaged portion of an image. Selected cropped image (b) is highlighted in red into the original image (a). Pixel position map is detected (c) and image restored (d) by using the algorithm in [15] (photo courtesy of Alinari Archives in Florence).

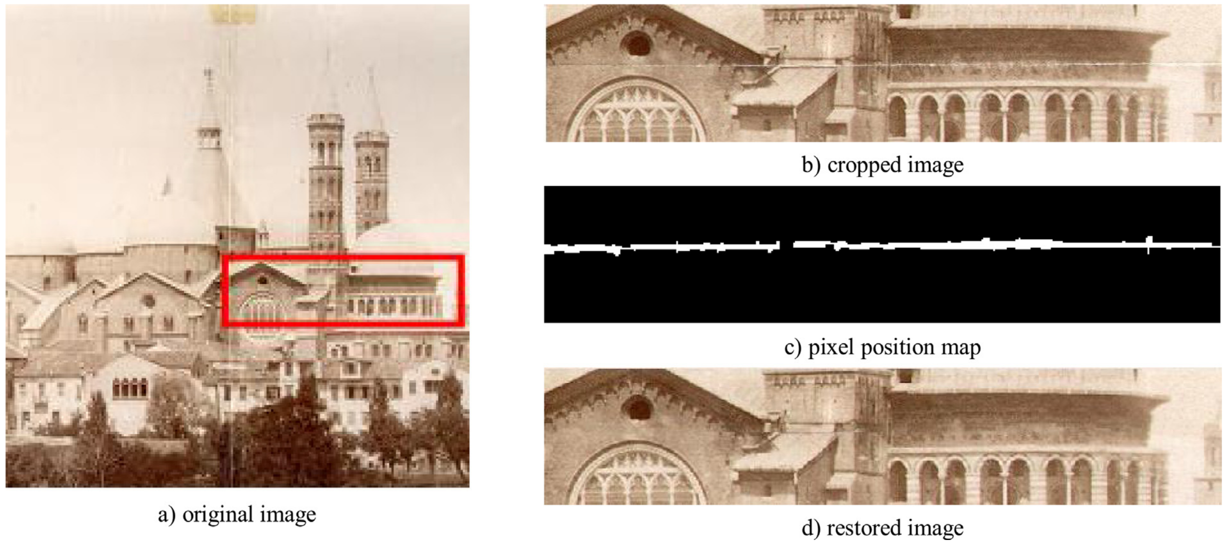


Fig. 11. Another example of a restoration process. Selected cropped image (b) is highlighted in red into the original image (a). Pixel position map (c) is detected by using the algorithm of Ardizzone et al [20]. Cropped image is restored (d) by using the Criminisi [30] inpainting algorithm (photo courtesy of Alinari Archives in Florence).

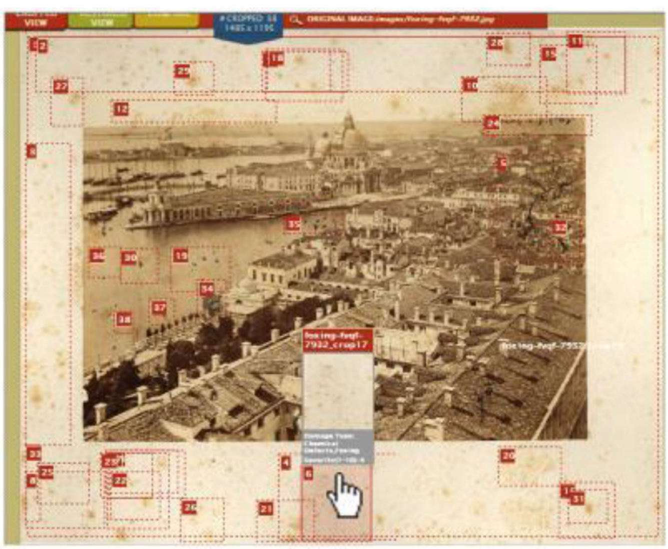


Fig. 12. Restoration browser: cropped view. All the defects of the original photo which had been processed (classified, detected or restored), are shown in this view. Passing the mouse pointer onto these areas, the tool shows information about the damage by which that crop is affected.

Fig. 13. Restoration browser: restored view (zoom) All the damaged parts which had been restored are highlighted. They are interactive: passing the mouse pointer onto these areas, it is possible to see their restored versions, and to have information about the applied restoration methods.

by the system, also avoiding the need to train the classifier after each use.

This solution implements the *similarto* property of the proposed ontology, and achieved around the 72% of accuracy. It is clear that this percentage can be improved if more advanced classifiers are used. The definitive solution of the problem should be a classification box which would combine standard and damage specific descriptors.

Once the cropped image is classified, the proper restoration "path" has to be applied. The current version of the tool let the user to choose between three options:

- Using one of all the existing restoration paths in the DB. The user selects one of the restoration paths in the list and, when needed, manually sets the required input parameters.
- Creating a new restoration path, combining existing algorithms. At the end of the process the new restoration path is saved as a new restoration type, and its implementation, as a sequence of applied algorithms, recorded in the DB. A new path is now at disposal for the next restorations.
- Allowing the system to suggest the appropriate restoration paths for the current image.

The last option needs a further explanation. Our tool analyzes all the cropped images which are affected by the same damage of the current image, and which has been already restored. Then it proposes to the user those restoration paths from the DB which have been yet used to restore these images (see bottom part of Fig. 9). Paths in the list are ranked, according to the mean score of the previous restorations' results. The user can select one of the elements of this list, and manually set the required input parameters, when needed. Otherwise, the tool can suggest the best parameters to the user, comparing the current image to the subset of images,

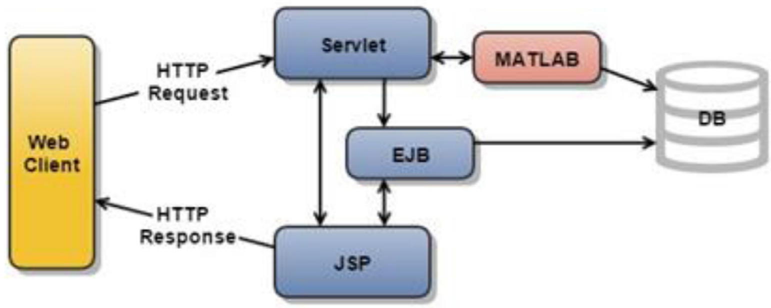


Fig. 14. Architecture of the web application.

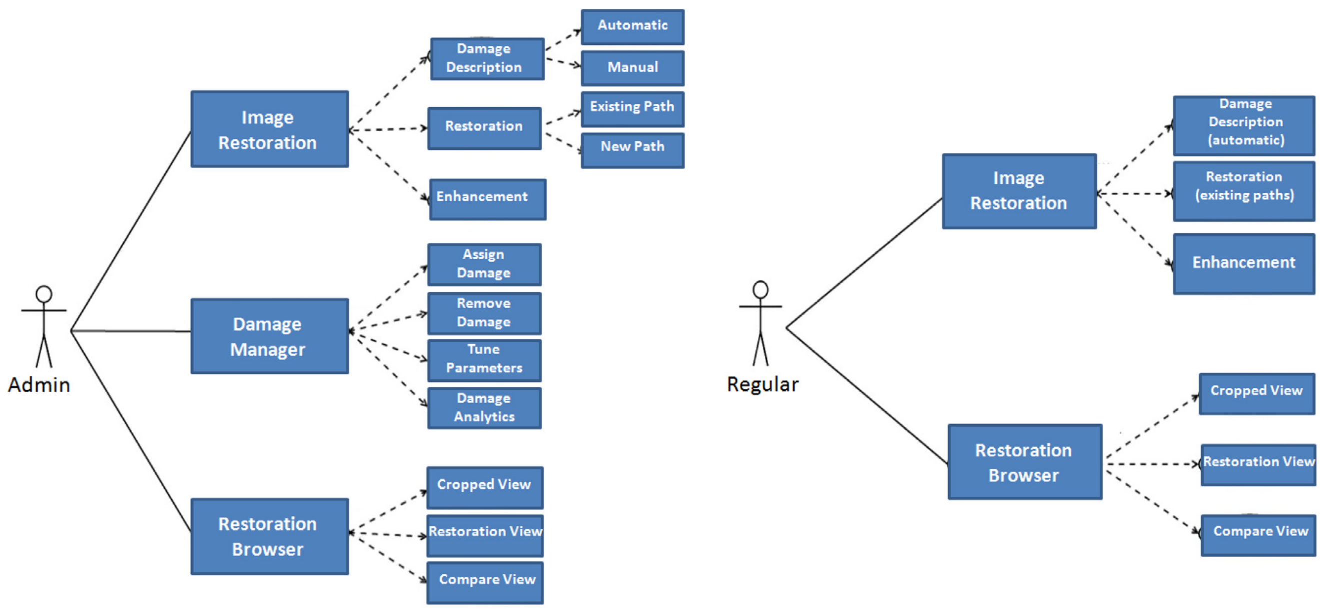


Fig. 15. Admin vs regular users.

affected by the same damage, which has been previously restored by the chosen path. It can be reasonably supposed that similar images, according to the proper descriptors, can be processed using the same parameters. In some cases the chosen restoration path requires a binary mask, into which pixels of the crop are labeled as damaged or uncorrupted. Whenever an automatic defect detection is not a part of the restoration path, another tool is provided to the user, to let him manually select the pixels to restore and to create the required binary mask. This is only of one the possible combinations of detection-restoration path in our database, but is the best for that specific damaged image. Figs. 10 and 11 show two examples of restoration paths, applied to images with different defects.

6.1. Algorithms

To evaluate our proposed architecture in real cases, we implemented several algorithms for our tool, in order to be part of the restoration process of our degraded images. As described in Section 4.1, we have 5 different subclasses of algorithms: Selection, Description, Detection, Restoration and Enhancement. We implemented 3 different Selection algorithms: rectangular cropping, polygonal cropping and magic wand, which is based on a simple region growing process. As Description algorithms we used some standard color descriptors (Dominant Color Descriptor and Color Layout [31], Color Coherence Vector [32]) and texture descriptors (Edge Histogram and Gabor Filters [31]), and some specific descriptors (Foxing [33]). We also implemented 3 Detection algorithms (Foxing and Blotches [15], Scratches [20]), 3 Restoration algorithms for specific defects (Foxing[15] and Blotches [15,16], Scratches [20]), and two generic inpainting algorithms (Criminisi [30], Wei-Levoy [34]). We further included in our knowledge base some user guided Enhancement algorithms (Contrast Enhancement, Brightness Adjustment, Custom Mask Spatial Filtering, Histogram Equalization [27]) and an unsupervised Defading algorithm [17]. Adding a new algorithm to the tool is very simple: it requires just putting the file in the proper directory (the script must include in the first lines a comment to explain the input parameters of the function) and updating the related Database table.

6.2. Restoration browser tool

Our software has another main functionality: users may visually represent the content of the database, by using the Restoration Browser tool. The Restoration Browser tool can extract relevant information from the database, and present them to the user as an interactive XHTML page (see Fig. 12). Three different “views” are allowed:

- Cropped view (Fig. 12): to show all the damaged crops of an original image. Information about each single area can be shown by passing mouse pointer on the corresponding highlighted zone.

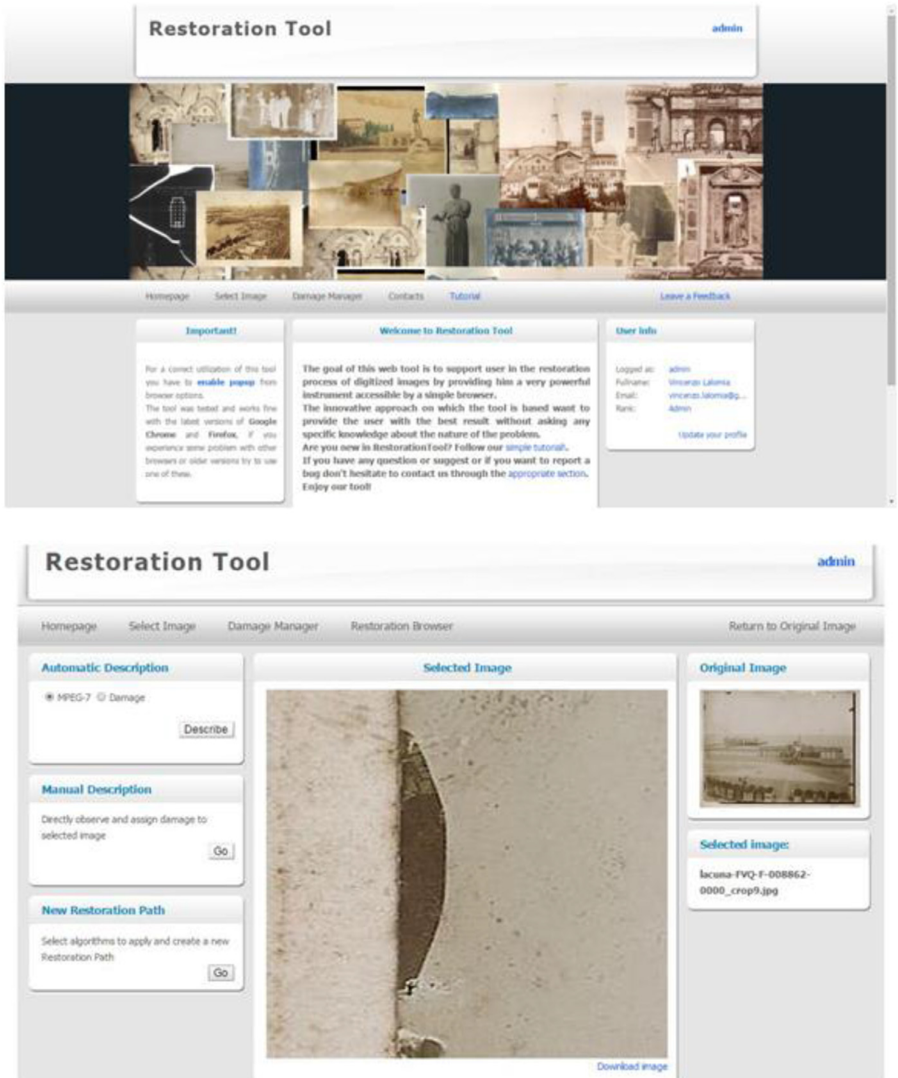


Fig. 16. Some screenshots of our restoration tool: home page (top), description (bottom).

- Restored view (Fig. 13): to show all the restored parts of an image. Visual improvement obtained by applying restorations can be seen using the mouse pointer:
- Compare view: to compare cropped images before and after restoration, and the original image with the corresponding enhanced

All the views are linked together by clicking on highlighted areas. Moreover, many other visual reports can be created, by querying, on cropped images affected by a specific damage, images restored by a specific path, etc.

7. Testing and validation

Once we created the knowledge base, we implemented a web application to test our system, and we involved several students as testers to evaluate the potentials of the proposed architecture.

7.1. Dataset

Our dataset is made of 20 images coming from the Alinari Archives in Florence, composed of high resolution, color, b/w digital images, digitized from original pictures, diapositives and slides. Photos have been acquired at a resolution such that digitization error is negligible. Nevertheless the analysis of the digitization error after the acquisition phase is outside of the scope of our work. Starting from these photos we created 250 damaged crops of a subset of the categories of Table 2. In general, all the local defects (spots, cracks, lacking portions, etc.) can be corrected with a combination of a manual selection and an inpainting algorithm, but in the current version our system does not have specific detection and restoration algorithms for all the local defects. For the global defects, our system includes a defading algorithm, while there are no algorithms to correct distortions in the chromaticity (like yellowing). We restored all the defects in our dataset with the available algorithms, to create our knowledge base.

7.2. Web application

To test our framework we implemented a Web Application version of our system, named “Restoration Tool”. The tool has been implemented as a Client-Server application, which uses J2EE, MySQL, XHTML, JSP and Matlab. Fig. 14 shows the Architecture of the Web application. It allows the users to have a powerful online tool to restore their photo, simply usingtheir browsers. The users can upload their photos to the systems and can exploit the knowledge based restoration functions of the tools. The processing is “transparent to the users”, as it is executed by the server side.

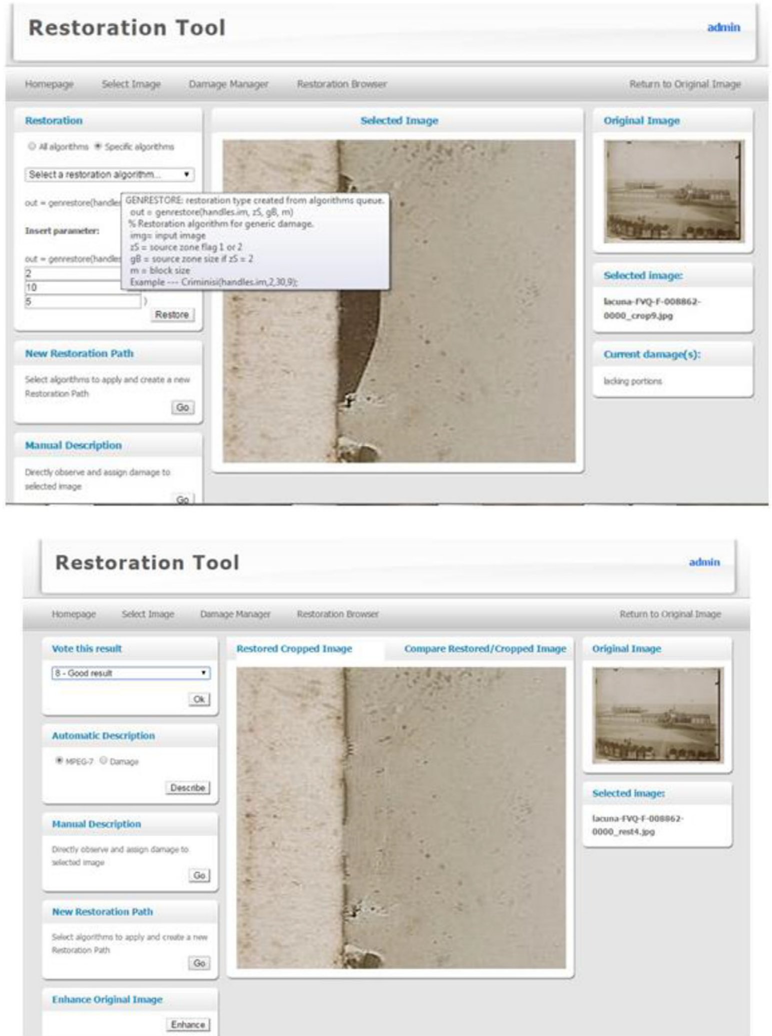


Fig. 17. Some screenshots of our restoration tool: applying algorithm (top), evaluation (bottom).

The Restoration Tool implements all the functionalities of the prototype, but with a significant change. The knowledge base has been modified adding the "user" entity, in order to manage the accesses to the Web site. Furthermore each user has a "role" that gives to him/her different rights to the several functionalities of the tool. The Admin user can access to all the functions without restriction (see Fig. 15). Regular users can use only the functionalities that do not modify the knowledge base: they cannot use the manual damage assignment tool; they cannot create new restoration paths; they cannot access to images uploaded by other regular users; they can work only on the images they uploaded or with common images (the original images of our dataset, that we made available for testing).

Figs. 16 and 17 show some screenshots of our web application.

7.3. Experimentation

To test our system we asked to 21 students, with different academic backgrounds, to use our application and to leave us a feedback, compiling a questionnaire, which is divided into two parts. The first part is proposed to evaluate the technical skills of the testers (computer, Internet, photo editing. see Table 3). The second part is used to evaluate the tool (interface, ease of use, results, etc. see Table 4).

Table 3
Skills evaluation. E = Expert, G = Fairly Good, A = Average, U = Unskilled.

Questions / Answers	E	G	A	U
How do you rate your computer skills?	43%	19%	33%	5%
How do you rate your internet skills?	29%	29%	33%	10%
How much are you skilled in photo-editing/virtual restoration software?	5%	33%	43%	19%

Regarding Table 3, note that only about 60% of the testers affirm to have computer skills over the average, while about 40% of the testers have average or no skills on photo-editing. From Table 4, it is evident that all the testers stated that the application is attractive and the functions are easy to find (about 40% SA and about 60% A). In relation to other commercial photo-editing software, the proposed tool is easy to use (about 30% SA and about 50% A), and it does not require too many steps to restore the photo (about 40% SD, about 60% D). Furthermore, by disaggregating data, all the users who are expert in photo-editing agreed with the ease of use of our tool with respect to other commercial software. This means that they recognized that restoring a photo with our tool is easier and faster than with other tools. In fact, our tool requires a very limited interaction with the user (only the selection of the area to be restored), while the rest of the process is strongly guided by the system: no training time is needed to learn how to use it. Finally,

Table 4
Tool evaluation. SA = Strongly agree, A = Agree, N = Neither agree or disagree, D = Disagree, SD = Strongly disagree.

Questions / Answers	SA	A	N	D	SD
The application has an attractive presentation.	43%	57%	0%	0%	0%
In relation to other software I usually use, I found the application easy to use.	29%	48%	24%	0%	0%
I found the various functions in the system were well organized and easy to find.	38%	57%	5%	0%	0%
I think I would need the support of a technical person to be able to use the application.	0%	10%	29%	48%	14%
There are too many steps required to get some result.	0%	0%	0%	57%	43%
The results obtained by the tool are satisfying.	48%	52%	0%	0%	0%
I think I would like to use this system frequently.	24%	71%	5%	0%	0%

the results are very satisfying (about 50% SA and about 50% A) and the testers would like to use the tool frequently (about 25% SA and about 70% A).

In terms of efficiency, it is very difficult to give objective metrics about the time to restore a degraded image, for both our tool and the commercial ones. Execution time depends on the damage severity of the image (e.g. how many defects are included into the image), and on the type of defects the image is affected (e.g. which algorithm to use to correct these defects). Therefore we decided to show only a subjective evaluation of the ease of use of our tool with respect to the commercial ones, as shown in Table 4.

8. Conclusions

People wish to preserve their memories of the past, such as old photo albums, but physical supports are fragile. The first step to preserve old photos is to make them digital. In this paper we presented a new restoration model for digitized old photos, inspired by the restoration process of manual restorers for printed photos. The classical model is not suitable for our goals, as it typically deals with global degradations, while defects coming from digitization of printed photos are, in most cases, local.

We defined an ontology to represent objects which are involved in a restoration process, their properties and relationships between them. We also defined a set of rules, to describe actions to make through the process. The whole system is implemented within a knowledge-based framework, and it is much more than a simple collection of restoration methods: it aims to free users from the annoying task of image analysis, defect detection and inspection, choice of the best restoration paths to correct a specific defect.

The use of a knowledge base make our system able to: automatically annotate damaged images; create relationships between defects and restoration paths; suggest to non-expert users the best restoration path for each damaged image; create an intuitive visual representation of the restoration process, retrieving information from the database. Furthermore it is dynamic: it grows gaining experience from the analysis of new kind of defects, the description of new cropped images, and the application of new restoration methods.

Finally, non-expert users do not need to have skill about restoration: they do not care about which restoration method should be used to correct defects of their personal photos. Our system uses knowledge from past restorations, and improve its ability every time a new restoration is applied. Our tests, in fact, confirmed that our tool is easy to use, and requires less steps to restore a photo, with respect to other commercial photo-editing software. Moreover it gives to expert users the possibility to create new restoration paths, combining detection, description and restoration algorithms from the database.

Actually we are studying a solution to implement our standalone framework as a distributed architecture of services, with the goal to create a network of restoration experts, which will offer services and support, e.g. via web, to people who are interested in restoring their old photos.

Acknowledgements

This work has been partially funded by the MIUR (Italian Ministry of Education, University and Research) project FIRB 2003 D.D. 2186 - Ric December 12th 2003. The authors wish to thanks the Alinari Photo Archives in Florence for having permitted the use of their photos in this research. We also acknowledge all the students who contributed with their work to the implementation of the restoration tool and its plug-ins, and in the testing phase.

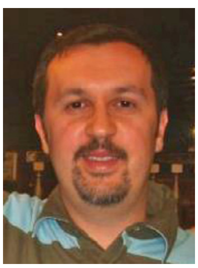
References

[1] G.V. Landon, A pipeline for digital restoration of deteriorating photographic negatives, Int. J. Digit. Libr. 16 (1) (2015) 79–88.
[2] A. Del Mastio, V. Cappellini, R. Caldelli, A. De Rosa, A. Piva, Virtual restoration and protection of cultural heritage images, in: Digital Signal Processing, 2007 15th International Conference on, 2007, pp. 471–474.
[3] M. Cislariu, M. Gordan, A. Vlaicu, C. Florea, S.C. Suteu, Defect detection and restoration of cultural heritage images, Acta Technica Napocensis 52 (4) (2011).
[4] A. Maronidis, C. Voutounos, A. Lanitis, An integrated tool for virtual restoration of byzantine icons, information, intelligence, systems and applications (IISA), in: Fourth International Conference on, Piraeus, 2013, pp. 1–6.
[5] O. Samko, Y.K. Lai, D. Marshall, P.L. Rosin, Virtual unrolling and information recovery from scanned scrolled historical documents, Pattern Recognit. 47 (1) (January 2014) 248–259.
[6] R. Hedjam, M. Cheriet, Historical document image restoration using multispectral imaging system, Pattern Recognit. 46 (8) (August 2013) 2297–2312.
[7] R.F. Moghaddam, M. Cheriet, A multi-scale framework for adaptive binarization of degraded document images, Pattern Recognit. 43 (6) (June 2010) 2186–2198.
[8] N. Bhardwaj, S. Agarwal, V. Bhardwaj, An imaging approach for the automatic thresholding of photo defects, Pattern Recognit. Lett. 60–61 (1) (August 2015) 32–40 ISSN 0167-8655.
[9] L. Zhang, A.M. Yip, M.S. Brown, C.L. Tan, A unified framework for document restoration using inpainting and shape-from-shading, Pattern Recognit. 42 (11) (November 2009) 2961–2978.
[10] T. Schreiber, B. Dubbeldam, J. Wielemaker, B. Wielinga, Ontology-based photo annotation, IEEE Intell. Syst. 16 (3) (May/June 2001) 66–74.
[11] S. Jiang, J. Du, Q. Huang, T. Huang, W. Gao, Visual ontology construction for digitized art image retrieval, J. Comput. Sci. Technol. 20 (6) (2005) 855–860.
[12] F.S. Abas, K. Martinez, Craquelure analysis for content-based retrieval, in: Proc. 14th International Conf. on Digital Signal Processing, July 1-3, Santorini, Greece, 2002.
[13] T. Ruzic, A. Pizurica, Context-aware image inpainting with application to virtual restoration of old paintings, in: In IEICE Information and Communication Technology Forum (ICTF-2013), 2013.
[14] B. Cornelis, T. Ružić, E. Gezels, A. Dooms, A. Pižurica, L. Platiša, J. Cornelis, M. Martens, M. De Mey, I. Daubechies, Crack detection and inpainting for virtual restoration of paintings: the case of the ghent altarpiece, Sig. Process. 93 (3) (March 2013) 605–619.
[15] F. Stanco, L. Tenze, G. Ramponi, Virtual restoration of vintage photographic prints affected by foxing and water blotches, J. Electron. Imag. 14 (4) (Oct.-Dec. 2005).
[16] V. Bruni, A. Crawford, A. Kokaram, D. Vitulano, Semi-transparent blotches removal from sepia images exploiting visibility laws, Sig. Image Video Process. 7 (1) (2013) 11–26.
[17] G. Ramponi, V. Bruni, Virtual restoration of faded photographic prints, in: Proc. Of 14th European Signal Processing Conf., 4-8, Florence, Italy, Sept. 2006.
[18] A.M. Ahmed, Color restoration techniques for faded colors of old photos, printings and paintings, in: IEEE International Conference on Electro/Information Technology, (EIT 09), 7-9, June 2009, pp. 151–156.
[19] V. Bruni, D. Vitulano, A generalized model for scratch detection, IEEE Trans. Image Process. 13 (1) (January 2004).
[20] E. Ardizzone, H. Dindo, G. Mazzola, Multidirectional scratch detection and restoration in digitized old images, J. Image Video Process (2010) Article 8.
[21] R.C. Chang, Y.L. Sie, S.M. Chou, T.K. Shih, Photo defect detection for image inpainting, in: Multimedia, Seventh IEEE International Symposium on, 2005, pp. 403–407.

[22] G. Maino, M. Monti, Color management and virtual restoration of artworks, in: Color Image and Video Enhancement, 2015, Springer International Publishing, 2015, pp. 183–231.
[23] P. Nemes, M. Gordan, A. Vlaicu, Color restoration in cultural heritage images using support vector machine, Scientific Computing and Cultural Heritage, Springer Berlin Heidelberg, 2013, pp. 51–59.
[24] A. De Polo, F. Stanco, G. Ramponi, Towards the automated restoration of old photographic prints: a survey, in: Proc. of IEEE EUROCON 2003, Ljubljana, Slovenia, September 2003, pp. 370–374.
[25] E. Ardizzone, H. Dindo, U. Maniscalco, G. Mazzola, Damages of digitized historical images as objects for content based applications, in: Proc. European Conference on Signal Processing, EUSIPCO 2006 Firenze, Italia, September 4 - 8 2006 (pp. CD-ROM).
[26] E. Ardizzone, A. De Polo, H. Dindo, G. Mazzola, C. Nanni, A dual taxonomy for defects in digitized historical photos, in: International Conference on Document Analysis and Recognition (ICDAR 09), 2009, pp. 1166–1170.
[27] R.C. Gonzales, R.E. Woods, Digital Image Processing, second ed., Prentice Hall, 2002.
[28] http://protege.stanford.edu/.
[29] http://www.w3.org/Submission/SWRL/.
[30] A. Criminisi, P. Perez, K. Toyama, Region filling and object removal by exemplar-based inpainting, IEEE Trans. Image Process. 13 (9) (2004).
[31] T. Sikora, The MPEG-7 visual standard for content description - nn overview, IEEE Trans. Circ. Syst. Video Technol. 11 (June 2001) 696–702 special issue on MPEG-7.
[32] G. Pass, R. Zabih, J. Miller, Comparing images using color coherence vectors, in: ACM Conference on Multimedia, Boston, Massachusetts, Nov. 1996, pp. 65–74.
[33] E. Ardizzone, H. Dindo, G. Mazzola, A set of low-level descriptors for images affected by foxing, in: In Signal Processing Conference, IEEE, 2008, pp. 1–5. 16th European.
[34] L.Y. Wei, M. Levoy, Fast texture synthesis using tree-structured vector quantization, in: Proceedings of the 27th annual conference on Computer graphics and interactive techniques, ACM Press/Addison-Wesley Publishing Co, 2000.

Edoardo Ardizzone is a Full Professor of Computer Systems at DIID, Università degli Studi di Palermo, Palermo, Italy, where he currently "Image Processing" at the graduate course of Computer Engineering. His research interests include image processing and analysis, medical imaging, image restoration and content-based image and video retrieval.

Haris Dindo is a Research Scientist and Assistant Professor at DIID, Università degli Studi di Palermo, Palermo, Italy, where he earned his Ph.D. in Computer Engineering in 2007. Actually he is mainly investigating applications of advanced forms of social learning and verbal interaction in robotics.



Giuseppe Mazzola received his PH. D. in Computer Engineering from Università degli Studi di Palermo, Palermo, Italy, in 2008. He is currently a temporary research fellow at, at the same university. His research interests include: Digital Image Forensics, Image Restoration, Visual Saliency, Content-Based Multimedia Retrieval, Geographical Information Retrieval.