Image Analysis in Technical Documentation (Discussion Paper)

Fabio Carrara[1], Franca Debole[1], Claudio Gennaro[1], and Giuseppe Amato[1]

Institute of Information Science and Technologies (ISTI), Italian National Research Council (CNR), Via G. Moruzzi 1, 56124 Pisa, Italy
name.surname@isti.cnr.it

Abstract. In the era of Big Data, manufacturing companies are overwhelmed by a lot of disorganized information: the large amount of digital content that is increasingly available in the manufacturing process makes the retrieval of accurate information a critical issue. In this context, and thanks also to the Industry 4.0 campaign, the Italian manufacturing industries have made a lot of effort to ameliorate their knowledge management system using the most recent technologies, like big data analysis and machine learning methods. This paper presents the on-going work done within the ADA project, with special emphasis on the specific image analysis work carried out to extract information from images contained in the so different document of the manufacturing companies, partners of the project.

Keywords: Image Analysis · Machine Learning · Big Data · Manufacturing companies

1 Introduction

Manufacturing companies, which produce complex products and manage large plants, generate a consistent flow of data and information throughout the company processes, from acquisition to production and maintenance of the products themselves. In this amount of data, a significant part consists of texts, graphics, and images obtained as a transposition of the know-how of human personnel.

Collecting and retrieving all this data and information quickly and easily is vital for speeding up internal business activities. For example, during the design phases of a new product, it is useful to be able to identify, in past projects, specifications, data and information contained in lessons learned, in risk analysis, to carry out more reliable and innovative design activities. In case of plants maintenance, it is invaluable for operators to have immediate access to the information necessary to carry out their work quickly and effectively.

The needs of the manufacturing companies described above, however, clash with the complexity of knowledge management, due both to the large quantity and to the heterogeneity of the data and documents to be processed: the technological tools currently available on the market are not able to rise this challenge and effectively meet these needs. In this context, therefore, the main target of the ADA project is to design and develop a platform based on big data analytics systems that allows for the acquisition, organization, and automatic retrieval of information from technical texts and images in the different phases of acquisition, design & development, testing, installation and maintenance of products.

In this paper, we illustrate the work carried out in the ADA project focusing on the image content retrieval part: the images contained in the corporate documents constitute a relevant source of information that could be relevant in the manufacturing work flow. On this context, we developed specific techniques for the extraction, classification, recognition, and tagging of images within technical documentation. The architecture and the methodologies of our work are presented on Section 3 and Section 4, respectively.

2 Related Work

Image recognition techniques have been extensively studied in the last decade in the field of computer vision and the recovery of multimedia information. Deep learning techniques, such as those based on Convolutional Neural Networks (CNNs), represent today the state of the art for the most varied computer vision activities such as image classification, image recovery and object recognition [4]. Furthermore, the use of intermediate layer activation as a high-level descriptor (feature) of visual image content has become very popular and has proven to be effective as demonstrated by many scientific papers [11]. Convolutional neural networks exploit the computing power provided by the GPU-based architectures, in order to learn from huge collections of multimedia information (e.g. images). One of the limitations of this approach is that many collections of images available were created for academic purposes (e.g. ImageNet [9]) and can not be used effectively for applications such as those discussed in the ADA project.

In the project, the development of tools to search for graphic symbols belonging to technical schemes within the technical documentation is of particular importance. To this end, some works [5, 10] that try to tackle the problem by using CNNs seem to be promising. However, as Elyan et al. claim in [5], the application of CNNs to detect and localize symbols in drawings is still a challenging task. This is probably due to the complexity of the problem and also to the lack of sufficient annotated examples or publicly available data sets.

Elyan et al. [5] presented a semi-automatic and heuristic-based approach to localise symbols within engineering drawings, and then applied a CNN to classify the detected symbols. Similarly, Quan et al. [10] used an AlexNet to classify point symbols in color topographic maps.

3 Image Analysis Component

The technical documents of manufacturing companies often contain a large number of heterogeneous images such as graphics, wiring diagrams, mechanical drawings, etc. While on the one hand the images are almost always accompanied by text such as captions, descriptions, and labels, being able to recognize their content without using text is an increasingly requested feature: images represent a rich source of information.

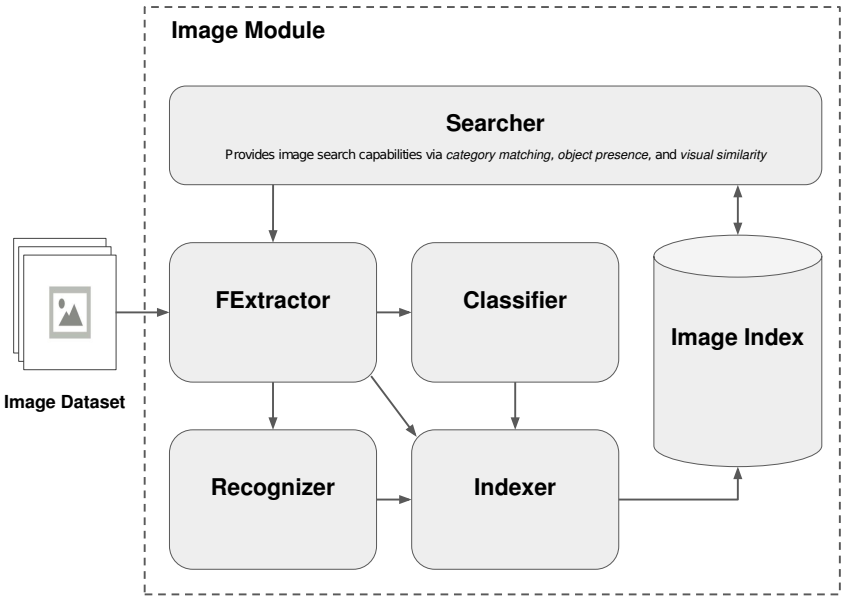

Fig. 1. A general view of the Image Analysis Module architecture.

The Image Analysis Module (Fig. 1) aims to enable companies to extract information from images contained within their technical documentation. This information is necessary to perform a search and classification of images based on visual content. The module in particular deals with the identification and extraction from the images the content, classifying them, recognizing some graphic elements, such as symbols, within them and looking for similar ones.

The Image Analysis Module is composed by the following sub-components:
FExtractor Module. This component is intended to extract from one image one or more visual descriptors (features) that allow one to perform searches based on visual similarity using an image as a query.

Classifier Module. The classifier component deals with the classification of the images in the various possible types in the field of technical and patent documentation (e.g. technical drawing in perspective, sections, electronic circuit, flow chart, etc.). The tool relies on modern Machine Learning techniques based on Deep Learning. The automatic classification is based on training a neural network on a number of examples for each of the types of images to be classified. The accuracy of the tool is strictly correlated to the number of training examples provided.
Recognizer Module. This component allows the automatic recognition of objects within the images classified by the classification tool: once the images are classified through the classification tool, the objects are automatically recognized within the image. For each type of image, the tool will have a number of examples related to the objects to be identified. Another result of this component is to automatically associate one or more tags with an image: the tags can be further enriched or corrected by analyzing the text in the document that refers to the image itself.
Indexer Module. This component deals with the appropriate indexing of both the visual features of the images and the context features related to the image: the image is decomposed into a visual part and into a symbolic part. For each image in fact, we will have both the global visual features necessary to perform visual similarity queries, and local features to perform queries able to detect and localize specific symbols contained in the image.
Searcher Module. This module deals with sorting the various types of search supported (see the details below).

The search component allows the user to perform different types of searches using the index created by the Indexer module:

1. Textual Search on the text correlated or extracted from the images.
2. Similarity Search on images: search by similarity of the basic elements (symbols) in an image archive using an example as a query.
3. Search for the categories to which an image belongs.

Furthermore, the Searcher Module is able to automatically handle external queries, i.e. using images that are not present in the index, as well as internal queries, i.e. using images already present in the database as queries. For the external queries, the Searcher Module uses FExtractor Module to extract features from the query image.

In the next section, we will describe the specific methodologies exploited for the realization of the four main mentioned modules: FExtractor, Classifier, Recognizer, and Indexer.

4 Methods

Due to their astonishing effectiveness on perceptual tasks, we resort to state-of-the-art Deep Learning techniques based on Convolutional Neural Networks

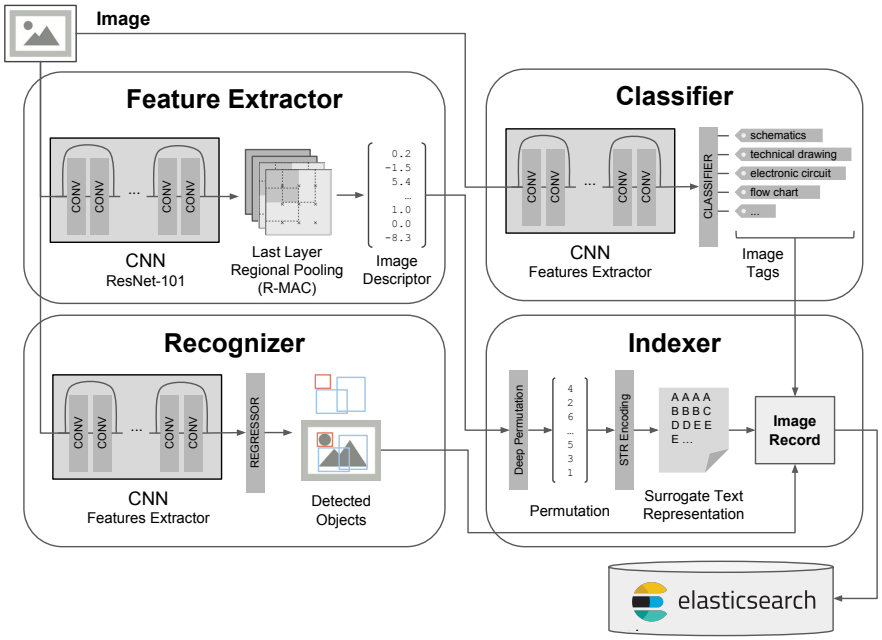

Fig. 2. Image Analysis Module: actual sub-components and their interaction.

(CNN) to implement the visual analyses conducted by the aforementioned modules.

In the following, we will describe in details the methods chosen to implement the main functions of the Image Analysis Module, and how they are distributed among its sub-components. As depicted on Figure 2, for each image on the dataset:

- we extract the features (FExtractor) as descriptors of the image on the index (Indexer); the extracted features are also made available to the Recognizer and Classifier modules;
- for specific categories, we use ad-hoc techniques for the object recognition (Recognizer) and the classification of the image (Classifier); we rely on simpler techniques based on previously extracted features otherwise;
- all the information deriving from FExtractor, Recognizer and Classifier are memorized in the Image Record on a specific index (Indexer).

4.1 Visual Similarity Search

For the implementation of the visual similarity search, we employ state-of-the-art global image descriptors extracted from CNN specifically tailored for image retrieval. Specifically, we select the Region Maximum Activation of Convolution (R-MAC) descriptors [14] as a compact and expressive image descriptors, which

enables our system to support instance-level visual object retrieval on both natural photos and schematics/synthetic images.

Given an image, the **FExtractor** module first feeds it to a pre-trained convolutional network to extract the output of the last convolutional layer, which is composed by multiple feature maps having two spatial dimensions. Then, we compute the R-MAC descriptor by pooling and aggregating different parts (or regions) of those feature maps. The feature maps are max-pooled over several regions on their spatial dimensions, and the obtained vectors are aggregated by summation and the result l2-normalized.

The choice of the pre-trained CNN for the extraction of the feature maps is essential: instead of choosing a network trained on generic object recognition tasks (such as ImageNet), we select the model developed by [6], a ResNet-101 convolutional network trained specifically for the task of same-object retrieval. With this configuration, we obtain 2048-dimensional image descriptors that can be compared with the cosine similarity to search for visual matches.

As already mentioned on Section 3, part of the work done for the ADA project is to make use also of the text correlated or extracted from the image. In the following paragraph, we will explain how we realized an appropriate index supporting both the visual and textual features of the images in an efficient way.

4.2 Indexing of Image Descriptors

While textual information, such as labels and tags, can be stored in a relational database, we need a similarity search index structure to store high-dimensional image features and efficiently compute the cosine-similarity to perform retrieval. Despite open-source indexes for efficient similarity search exist [8], they still come with caveats and, in general, are not mature and well-supported as the scalable disk-based textual indexes such as Elastichsearch.

Seeking for simplicity, we propose to adopt Elasticsearch in the **Indexer** and **Searcher** modules as database for all the information extracted by the visual analysis modules, and adapt our similarity search needs to the full-text search engine provided by the software. Specifically, the adoption of two techniques, *Deep Permutations* [3] and *Surrogate Text Representation* [1, 2], permit us to represent our image descriptors as text, index them using a full-text search engine based on inverted indexes, and perform similarity queries without the need of a image-specific similarity index.

The **Deep Permutations** technique is based on the fact that in high-level deep-learned features, each dimension represents an abstract visual concept, and its value specifies the importance of that concept. Similar features tend to have the same relative importance among visual concepts, and thus, we can approximate a float vector of features by sorting its dimensions in descending order and keeping the sorting permutation (an integer vector). By truncating the permutation to the top elements only, Amato et al. [3] demonstrated that this coarse approximation is sufficient to obtain state-of-the-art trade-offs between efficiency and effectiveness, permitting large-scale searches with query time in the order of seconds.

The **Surrogate Text Representation** technique aims at encoding a given integer vector in the term frequency values of a full-text search engine by generating an appropriate textual string. Used in combination with Deep Permutations, it permits us to store sparse truncated permutation in the inverted index for textual data and leverage the vector document model of full-text search engines to perform cosine similarity searches.

4.3 Image Categorization and Object Detection

For certain documentation, the **Classifier** module needs to assign each image to one or more categories defined by the system users. Thus, we have to implement a multi-class multi-label image classifier. For the categories with a considerable number of training examples, a CNN can be trained to implement the image classifier. Instead of defining and training a new model from scratch, we propose to leverage the *transfer learning* practice, exploiting the knowledge of popular CNNs that have been successfully trained in large-scale generic object recognition tasks. Specifically, we plan to use the ResNet-50 [7] model pre-trained on ImageNet1k, replacing the last classification layer to match our number of classes and to perform fine-tuning on the available training samples. When training samples are scarce, we resort to a simpler kNN classifier based on visual similarity: we reuse the same features extracted by the **FExtractor** module and the cosine similarity matching function in the implementation of the kNN classifier.

For specific categories of images, the **Recognizer** module needs to detect the presence of particular objects of interest. If the location of the object is not required, R-MAC features matching can be used to reveal object presence. Being an aggregation of local region descriptors, matching the R-MAC descriptor of a sub-region of an image against the whole image descriptor will yield a high score. Thus, we can perform object detection via instance-level similarity search reusing the same modules. Instead, if more fine-grained information about detected objects is required, such as the specific localization in the image, and enough training samples are available, we resort to more complex object detection techniques based on Deep Learning, such as region-based CNNs [13] or single-stage detectors [12], that can be fine-tuned on the specific object category to detect.

Both the information extracted by the **Classifier** and **Recognizer** modules are stored by the **Indexer** module in Elasticsearch as additional fields of the image record, and accordingly searchable through the **Searcher** module.

5 Conclusions

This paper briefly introduces the on-going work on image analysis in the ADA project, whose main aim is to support the innovation of the production process of the manufacturing companies. We focused on the description of the methodologies carried out to extract relevant information from images contained in the different document provided by the project partners.

Acknowledgments

This work was partially funded by "Automatic Data and documents Analysis to enhance human-based processes" (ADA), CUP CIPE D55F17000290009. We gratefully acknowledge the support of NVIDIA Corporation with the donation of a Tesla K40 GPU used and a Jetson TX2 board used for this research.

References

1. Amato, G., Bolettieri, P., Carrara, F., Falchi, F., Gennaro, C.: Large-scale image retrieval with elasticsearch. In: The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval. pp. 925–928. ACM (2018)
2. Amato, G., Carrara, F., Falchi, F., Gennaro, C.: Efficient indexing of regional maximum activations of convolutions using full-text search engines. In: Proceedings of the 2017 ACM on International Conference on Multimedia Retrieval. pp. 420–423. ACM (2017)
3. Amato, G., Falchi, F., Gennaro, C., Vadicamo, L.: Deep permutations: Deep convolutional neural networks and permutation-based indexing. In: International Conference on Similarity Search and Applications. pp. 93–106. Springer (2016)
4. Donahue, J., Jia, Y., Vinyals, O., Hoffman, J., Zhang, N., Tzeng, E., Darrell, T.: Decaf: A deep convolutional activation feature for generic visual recognition. arXiv preprint arXiv:1310.1531 (2013)
5. Elyan, E., Garcia, C.M., Jayne, C.: Symbols classification in engineering drawings. In: 2018 International Joint Conference on Neural Networks (IJCNN). pp. 1–8. IEEE (2018)
6. Gordo, A., Almazán, J., Revaud, J., Larlus, D.: End-to-end learning of deep visual representations for image retrieval. International Journal of Computer Vision **124**(2), 237–254 (2017)
7. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. arXiv preprint arXiv:1512.03385 (2015)
8. Johnson, J., Douze, M., Jégou, H.: Billion-scale similarity search with gpus. arXiv preprint arXiv:1702.08734 (2017)
9. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems. pp. 1097–1105 (2012)
10. Quan, Y., Shi, Y., Miao, Q., Qi, Y.: A combinatorial solution to point symbol recognition. Sensors **18**(10), 3403 (2018)
11. Razavian, A.S., Azizpour, H., Sullivan, J., Carlsson, S.: CNN features off-the-shelf: an astounding baseline for recognition. In: Computer Vision and Pattern Recognition Workshops (CVPRW), 2014 IEEE Conference on. pp. 512–519. IEEE (2014)
12. Redmon, J., Farhadi, A.: Yolo9000: better, faster, stronger. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 7263–7271 (2017)
13. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. In: Advances in neural information processing systems. pp. 91–99 (2015)
14. Tolias, G., Sicre, R., Jégou, H.: Particular object retrieval with integral max-pooling of cnn activations. arXiv preprint arXiv:1511.05879 (2015)