Knowledge-Based Systems 248 (2022) 108859

Contents lists available at ScienceDirect

Knowledge-Based Systems

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

Leveraging Knowledge Graphs and Deep Learning for automatic art analysis

Giovanna Castellano, Vincenzo Digeno, Giovanni Sansaro, Gennaro Vessio[*]

Department of Computer Science, University of Bari, Italy

ARTICLE INFO

Article history:
Received 15 November 2021
Received in revised form 14 April 2022
Accepted 16 April 2022
Available online 23 April 2022

Keywords:
Knowledge graphs
Artificial intelligence
Deep learning
Computer vision
Graph neural networks
Digital humanities
Fine arts

ABSTRACT

The growing availability of large collections of digitized artworks has disclosed new opportunities to develop intelligent systems for the automatic analysis of fine arts. Among other benefits, these tools can foster a deeper understanding of fine arts, ultimately supporting the spread of culture. However, most of the systems proposed in the literature are only based on visual features of digitized artwork images, which are sometimes only integrated with some metadata and textual comments. A Knowledge Graph (KG) that integrates a rich body of information about artworks, artists, painting schools, etc., in a unified structured framework, can provide a valuable resource for more powerful information retrieval and knowledge discovery tools in the artistic domain. To this end, in this paper we present $\mathcal{A}rt\mathcal{G}raph$:[1] an artistic KG based on WikiArt and DBpedia. The graph already provides knowledge discovery capabilities without having to train a learning system. In addition, we propose a novel KG-enabled fine art classification method based on $\mathcal{A}rt\mathcal{G}raph$, which is used to perform artwork attribute prediction tasks. The method extracts embeddings from $\mathcal{A}rt\mathcal{G}raph$ and injects them as "contextual" knowledge into a Deep Learning model. Compared to the state-of-the-art, the proposed model provides encouraging results, suggesting that the exploitation of KGs in combination with Deep Learning can pave the way for bridging the gap between the Humanities and Computer Science communities.

1. Introduction

In recent years, Knowledge Graphs (KGs), and their underlying semantic technology, have emerged as a powerful tool for describing real-world entities and their relationships [1]. They are increasingly used for many practical tasks, from recommendations [2] to query answering [3], and many KGs have been constructed and made publicly available, such as Google Knowledge Graph and Facebook Entity Graph. At the same time, the last decade has seen remarkable advances in Deep Learning approaches based on neural networks [4], which have led to building ever more accurate systems in a wide range of areas, especially Computer Vision and Natural Language Processing. Combining the expressiveness of KGs with the learning ability of deep neural networks promises to develop even more effective algorithms for many *downstream* tasks.

One of the many domains that can benefit from combining KGs and Deep Learning is the artistic one. Leveraging Deep Learning algorithms in this domain, particularly Convolutional Neural Network (CNN) models, has already proven effective in tackling several challenging tasks, from object detection in paintings to style classification [5]. And this success is mainly due to the growing availability of large digitized fine art collections, such as WikiArt. However, while promising, most of the existing solutions rely solely on the visual features that a CNN can automatically extract from digital images of paintings, drawings, etc. (e.g., [6–8]), and these features are rarely integrated with some metadata and textual comments to address multi-modal retrieval tasks (e.g., [9,10]). In other words, the dominant approach is mainly one based on perception and recognition. This inevitably leads to ignoring an enormous amount of knowledge – already available from disparate sources – relating to the "context" of each artwork. An artwork, in fact, is characterized not only by its visual appearance but also by various historical, social, and contextual factors that allow us to frame the artwork in a richer and more multifaceted scenario.

A promising way to harness this knowledge and improve the accuracy of art-based analytic systems is to encode the contextual information of the artworks into a KG and use representations learned from the graph as an additional input to a Deep Learning model. Indeed, having a knowledge base in which not only artworks but also a rich plethora of metadata, contextual information, textual descriptions, etc., are unified in a structured framework can provide a valuable resource for more powerful

[*] Corresponding author.
E-mail address: gennaro.vessio@uniba.it (G. Vessio).

[1] $\mathcal{A}rt\mathcal{G}raph$ and associated code are publicly available on https://doi.org/10.5281/zenodo.6337958.

https://doi.org/10.1016/j.knosys.2022.108859

information retrieval and knowledge discovery tools in the artistic domain. Such a framework would be beneficial not only for enthusiastic users, who can exploit the encoded information to navigate the knowledge base, but also especially for art experts, interested in finding new relationships between artists and/or artworks for a better understanding of the past and modern art.

To this end, in this paper we present *ArtGraph*: an artistic Knowledge Graph. The proposed KG integrates information collected by WikiArt and DBpedia, and encodes a broad representation of the artistic domain, with multiple metadata and relationships between artists and artworks. Furthermore, we propose a novel approach to inject contextual knowledge into a deep neural network to perform fine art classification. The method jointly leverages visual features and Knowledge Graph embeddings, and their combined exploitation improves performance over using visual features alone. This paper extends our preliminary work in this direction reported in [11].

The rest of this paper is structured as follows. Section 2 reviews related work. Section 3 presents *ArtGraph*. Section 4 describes the proposed classification method. Section 5 presents and discusses the experimental setup and the results obtained. Section 6 concludes the paper and describes the future developments of our research.

2. Related work

Traditionally, automatic art analysis has been performed using hand-crafted features fed into traditional Machine Learning algorithms, e.g. [12–14]. Unfortunately, despite the encouraging results of feature engineering techniques, early attempts soon stalled due to the difficulty of gaining explicit knowledge of the attributes to associate with a particular artist or artwork. This difficulty arises because this knowledge typically depends on an implicit and subjective experience that a human expert might find difficult to verbalize [15,16].

In contrast, several successful applications in a range of Computer Vision tasks have demonstrated the effectiveness of representation learning versus feature engineering techniques in extracting meaningful patterns from complex raw data. One of the first successful attempts to apply deep neural networks in this context was the research presented by Karayev et al. [17], which showed how a pre-trained CNN can be quite effective in attributing the correct school of painting to an artwork. Since then, many works have focused on using Deep Learning techniques based on single-input [18,19] or multi-input models [8] to solve artwork attribute prediction tasks based on visual features. Other directions that have attracted the interest of the community working on this domain are visual link retrieval [20,21], object detection [22–24], and near-duplicate detection [25].

However, the problem of predicting the attributes of an artwork using only visual information is very challenging. For this reason, researchers felt the need to use contextual information along with visual features. Some attempts have been made to express contextual information using Knowledge Graphs and inject this information into Deep Learning models. One of the first works, which inspired our research, is the *ContextNet* framework proposed by Garcia et al. [26]. They combined a multi-output CNN trained to solve attribute prediction tasks based on visual features with a second model, which is a simple encoder, based on non-visual information extracted from artistic metadata encoded using a KG. To encode the KG information into a vector representation, the popular node2vec [27] was adopted. The KG was built using only the information provided by the previously proposed SemArt dataset [10]. To do this, the authors defined a node for each artwork and connected each artwork to its attributes. They used some metadata, including artist, title, technique, etc. Also, by applying an *n*-gram model to the title, its keywords were extracted and added to the graph. Despite its richness, the graph constructed by Garcia et al. has two limitations. Metadata is only available for artworks in the dataset, so adding a new artwork would result in a lack of domain information about it. In addition, the proposed graph connects artworks with the same artist, but does not consider the relationships between artists, such as artistic influence. Our work is framed in the direction of overcoming these limitations. In particular, we propose the use of a source of knowledge external to the dataset, i.e. Wikipedia, which provides an enormous amount of information, even in a structured form.

Finally, to perform fine art classification, the most recent works [28–30] make use of KGs as input for Graph Neural Network models [31]. However, these works either use the entire graph before splitting the dataset, or generate pseudo-labels for the test instances so that an "extended" graph is obtained at test time. This strategy can improve performance, but we observe that it suffers from some drawbacks. First of all, the generation of pseudo-labels requires a re-training of the overall model, which prevents its use in real-time. Secondly, the construction of an extended graph based on pseudo-labels strongly depends on the distribution of the test data and, more precisely, to what extent they are linked to each other and the existing KG. This problem is taken to the extreme if the model is asked to predict the attributes of a single test instance one at a time. In other words, while successful, they adopt a transductive rather than an inductive approach. In this paper, we take an inductive approach and propose a model suitable for real-time scenarios, where nothing is known about a new artwork beyond what is "perceived" visually.

3. *ArtGraph*

Artworks cannot be studied based only on their visual appearance, but also considering other historical, social, and contextual factors. For example, it may be useful to know that the author of an artwork was a pupil of another artist, or that two artists adhered to the same artistic current. Such knowledge can be provided either through data in a structured form, such as metadata on authors or styles, but also in unstructured form, such as the description provided by an art expert or available on Wikipedia. In this view, a comprehensive KG would provide a more expressive and flexible representation of the relationships between entities relating to art, which cannot be obtained by considering only the visual content. To this aim, we developed *ArtGraph* as a KG in the art domain capable of representing and describing concepts related to artworks. A comparison between our proposed KG and the one presented by Garcia et al. [26] is provided in Table 1.

The starting point for the construction of *ArtGraph* was WikiArt, which contains more than 250,000 artworks. For each artwork, if available, WikiArt provides metadata such as the author, the date of creation, the gallery in which it is exhibited, the style, and so on. To download artwork images and metadata, we used the freely available API. WikiArt also provides information about the authors such as birth date, biography, and Wikipedia link. Since WikiArt does not provide rich information about the artists, the Wikipedia link was exploited to obtain artist metadata from DBpedia. To query DBpedia we used SPARQL, which is a query language capable of retrieving and manipulating data stored in the Resource Descriptive Framework. The metadata retrieved from WikiArt and DBpedia were placed in several interconnected comma separated value files, which have been processed for data cleaning and preparation using the Python pandas library.

Table 1
Comparison between our KG and the one proposed by Garcia et al. [26].

KG	# Nodes	# Edges	# Artists	# Artworks	# Relations	# Relations
					btw artworks	btw artists
ContextNet	33,148	125,506	3166	19,244	7	0
ArtGraph	135,038	875,416	2501	116,475	10	7

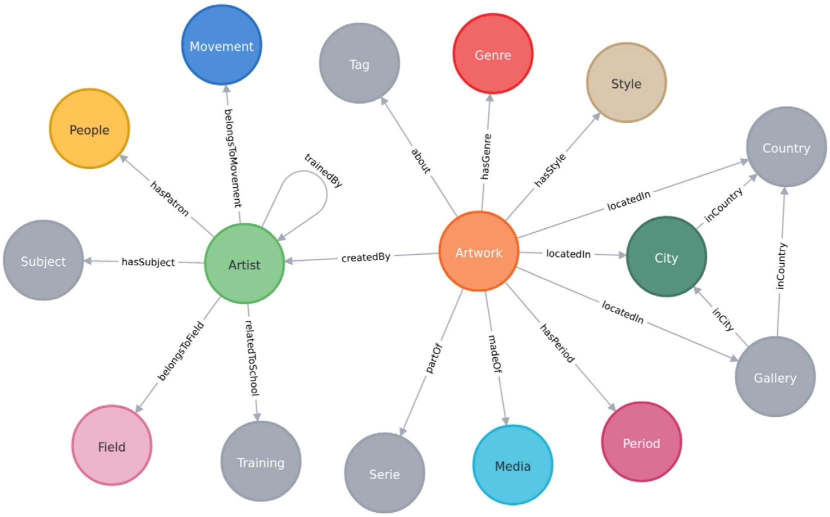


Fig. 1. Scheme of *ArtGraph*. The nodes correspond to relevant entities in the artistic domain, while the edges represent existing relationships between them.

Then, the KG was created using Neo4j (but many other graph databases could have been used the same way) and we modeled the graph as a *labeled property graph*. The conceptual scheme of *ArtGraph* is represented in Fig. 1. Each node type has the property *name*, which identifies the node instance. The *artwork* and *artist* nodes, which are more informative and contain more attributes, also contain some properties like the WikiArt url and the image url. Each artwork has only one associated digitized image, taken from WikiArt. The *artwork* node is connected to the nodes: *tags* (e.g., sea, birds), *genre* (e.g., portrait), *style*, *period*, *serie* (e.g., "The Holy Family''), *media* (e.g., canvas, oil), and the *gallery*, *city* and *country* where the artwork is located. The *artist* node is connected to the nodes: *field* (e.g., painting, illustration), *movement* (e.g., Art Déco, Surrealism), *subject* (e.g., people from Florence), the *training* school associated with the artist (e.g., Accademia di Belle Arti di Bologna), *people*, i.e. non-artist persons linked to the artist (e.g., Cosimo de' Medici), and other *artists*, who exerted an influence. Of course, there is finally a relation between *artworks* and *artists*. More details can be found on the publicly available repository.

This structure allows the creation of a network between artists, which is useful for further analysis even without explicitly training a learning system. In fact, it is possible to run queries that can be particularly useful for art analysis, such as: retrieving the direct and indirect influencing connection between artists with different degrees of separation; identifying artworks that are stored in a country other than those in which they were completed; retrieving all the works that are stored in a specific place; and so on (Fig. 2). In total, the resulting KG contains 135,038 nodes and 875,416 edges, with 2501 artists, 116,475 artworks, 18 genres, 32 styles, and many other metadata characterizing them.

4. Proposed classification method

ArtGraph encodes a valuable source of contextual knowledge that can be integrated with visual features automatically learned by deep neural networks to develop more powerful learning models in the art domain. Several tasks, in fact, could be addressed, such as artwork attribute prediction, multi-modal retrieval, and artwork captioning, which are attracting increasing interest in this domain [5]. Taking advantage of the developed artistic KG, we propose a new classification model that is used in this paper to predict the style and genre of a given artwork. The model, as shown in Fig. 3, is inspired by *multi-modal* and *multi-task* learning. In the following, we describe the three main steps of our proposed method:

- The multi-modal learning strategy adopted, i.e. the embedding generation;
- The multi-task strategy used to classify both style and genre at the same time;
- The specific approach taken at testing time, based on the projection of the visual features in the multidimensional space of the context features, i.e. the graph embeddings.

4.1. *Embedding generation*

Multi-modal learning aims to build models that can process and relate information from multiple modalities [32]. The underlying principle is that multiple modalities can provide different views of the same input which, when processed simultaneously, can help increase predictive accuracy. The most common method in practice is to combine several high-level embeddings from the same input by concatenating them and then applying a softmax, with the aim of transferring knowledge between modalities and their representations. In our case, we can exploit both the visual embedding obtained from a deep neural network aimed at extracting visual features from the digitized artwork and the corresponding graph embedding obtained from the KG.

As for the visual embeddings, the three-channel artwork images are resized to 224 × 224 and propagated through a Vision Transformer (ViT) [33] pre-trained on ImageNet and fine-tuned on our image dataset. We take the final output features after linear transformation which result in a visual embedding $\mathbf{h}_v \in \mathbb{R}^{768}$. ViT uses a Transformer-like architecture that represents

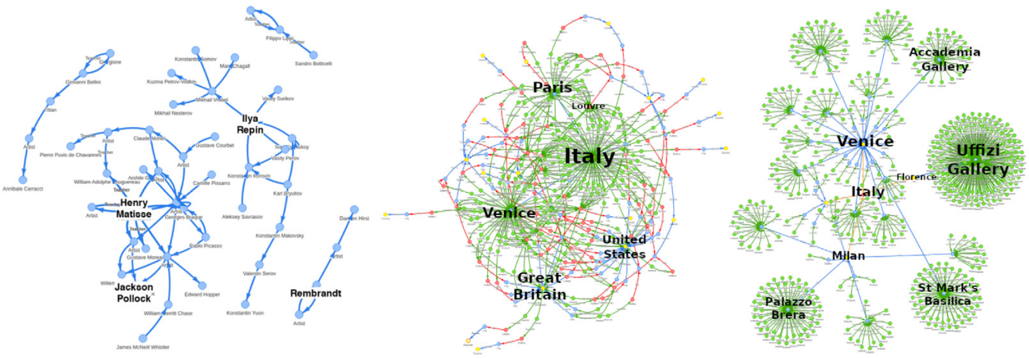


Fig. 2. From left to right, examples of query results: retrieving direct and indirect influences between artists; identifying artworks that are stored in a country other than those in which they were completed; retrieving all the works that are kept in particular places. The colors are automatically set by the Neovis.js visualization tool to reflect some properties of the sub-graph.

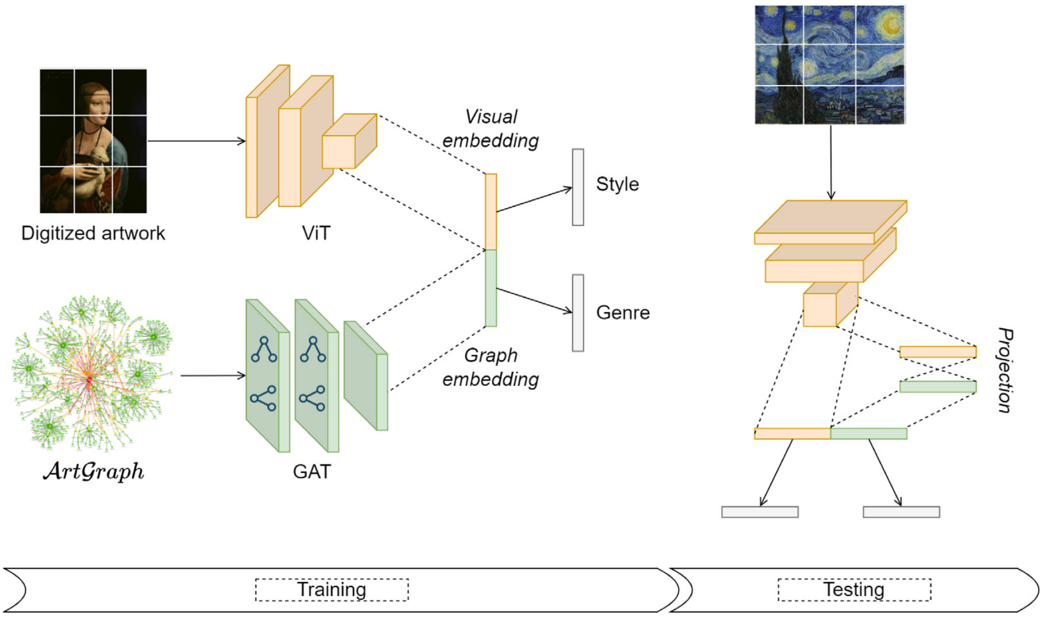


Fig. 3. Schema of the proposed multi-task multi-modal model. A concatenation layer receives both the contribution of visual embeddings, extracted from ViT, and graph embeddings extracted from the information encoded in the KG. The overall network learns to minimize the error made to predict the correct style and genre of a given artwork. At testing time, a projection function is used to project visual features in the same multidimensional space as contextual features, thus creating graph embeddings for unseen artworks.

an input image as a series of patches. These patches are then linearly embedded, added to position embeddings, and the resulting features are fed into a standard Transformer encoder. To perform classification, an additional learnable "classification token" is used.

As for the graph embeddings, these are extracted from *ArtGraph* using a Graph Attention Network (GAT) [34], which provides the context information for each artwork, aimed at improving the representativeness of the visual features. GATs use weights associated with each node in the neighborhood which tell how much "attention" should be paid to a message from that neighbor. The attention weights are learned during training. In particular, a GAT consisting of two layers with 128 hidden units each was used, with batch normalization, an attention head, and the ReLU activation function.

The motivation for using GATs instead of other node embedding algorithms, such as node2vec [27], was to take advantage of the heterogeneity of *ArtGraph* and the semantics of its relations. Also, with Graph Neural Networks, features can be assigned to nodes. Each artwork node was assigned the visual feature vector $\mathbf{h}_v$. The other nodes (e.g., media and movement) have been assigned an identity feature, specifically a one-hot indicator feature, which uniquely identifies that node. However, GATs cannot be trivially applied to heterogeneous graphs, since node and edge features of different types cannot be processed by the same functions. To get around this, and then be able to apply GAT to *ArtGraph*, we implemented the message and update functions individually for each edge type. Finally, to generate node embeddings, GAT was pre-trained to solve a node classification task, and the last hidden layer output was used as a node embedding $\mathbf{h}_g \in \mathbb{R}^{128}$. In other words, unlike node2vec, a supervised approach can be used. Since the attributes of interest are genre and style, each artwork will be assigned two node embeddings, one obtained from a genre classification task and the other from a style classification task. In the following, we describe how the overall architecture works.

4.2. Multi-task learning

As for the classification stage, instead of adding a single output layer and learning each classification task separately, we adopt a multi-task solution [35]. In particular, both visual and contextual features are combined by concatenation, $(\mathbf{h}_v, \mathbf{h}_g)$, and fed

Table 2
Single-task classification results.

Method	Style			Genre		
	Top-1	Top-2	Macro F1	Top-1	Top-2	Macro F1
ResNet	52.48%	70.91%	50.53%	65.76%	82.49%	58.61%
ViT	52.37%	70.46%	52.97%	65.92%	82.98%	61.62%
ContextNet [26] (ResNet+node2vec)	44.81%	64.16%	43.95%	60.44%	79.60%	55.80%
Our previous method [11] (ResNet+node2vec)	48.94%	67.83%	48.10%	63.64%	81.58%	58.24%
This work (ResNet+GAT)	54.80%	72.39%	52.02%	69.73%	84.25%	61.84%
This work (ViT+GAT)	58.31%	75.37%	56.32%	71.23%	85.70%	64.06%

Table 3
Multi-task classification results.

Method	Style			Genre		
	Top-1	Top-2	Macro F1	Top-1	Top-2	Macro F1
ResNet	48.46%	66.72%	47.23%	62.51%	81.12%	57.37%
ViT	53.09%	71.57%	52.58%	67.03%	84.96%	61.97%
ContextNet [26] (ResNet+node2vec)	42.61%	61.91%	41.42%	61.77%	79.87%	56.70%
Our previous method [11] (ResNet+node2vec)	48.20%	67.41%	47.86%	64.15%	82.35%	59.64%
This work (ResNet+GAT)	53.70%	71.90%	50.43%	70.13%	84.47%	61.73%
This work (ViT+GAT)	58.58%	76.13%	56.58%	72.29%	86.45%	65.29%

to a fully-connected head, one for each task (style and genre classification), with as many output units as there are classes to predict. In this way, features are shared between the tasks allowing the model to simultaneously exploit the semantic correlation between them. This strategy forces the model to learn visual features that share common elements between different contextual attributes of the artworks, so that the information about the style is useful for the classification of the genre, and vice versa. Furthermore, by propagating the error back to ViT, the model is forced to learn "context-aware" visual features that encapsulate some of the contextual knowledge in the graph.

Given a number of tasks T (two in our work, corresponding to the style and genre classification), and a number of classes C for each task, the loss function is:

$$\mathcal{L} = \sum_{i=1}^{T} \lambda_i \sum_{j=1}^{C} w_j \ell_c(y_j, \hat{y}_j),$$

where λ_i is a hyper-parameter that weights the contribution of each task i, while ℓ_c is the cross-entropy loss function defined as:

$$\ell_c(y_j, \hat{y}_j) = -y_j \log\left(\hat{y}_j\right),$$

where, for a given artwork, $\hat{y}_j$ is the softmax predicted output and y_j is the corresponding ground truth.

Finally, to counter the problem of class imbalance, each class in the loss function has been associated with a weight inversely proportional to its frequency. The aim was to penalize the misclassification made on the minority classes by setting higher class weights. The weight for the jth class was calculated as follows:

$$w_j = \frac{N}{C * n_j},$$

where N is the total number of artworks, C the number of classes and n_j the fraction of artworks belonging to class j.

4.3. Embedding projection

At testing time, it must be assumed that, for any new artwork, no knowledge is available other than its visual content, so it is not possible to know in advance the nodes of the graph to which the artwork can be linked. This is essential for simulating a real-time application of the model where the only information available is the digitized representation of the artwork in form of pixels. To help the model implicitly consider contextual knowledge, the main idea is to learn how to project the visual embeddings into the "context space" provided by the graph embeddings. This is done by learning a projection function $f : \mathbb{R}^{768} \rightarrow \mathbb{R}^{128}$ separately and using the projected features $\mathbf{h}_{\hat{g}} = f(\mathbf{h}_v)$ as test graph embedding. The function f is found by training a simple encoder which is asked to minimize a smooth-ℓ_1 loss function between the projected features $\mathbf{h}_{\hat{g}}$ and the graph embedding $\mathbf{h}_g$:

$$\ell_1\left(\mathbf{h}_{\hat{g}}, \mathbf{h}_g\right) = \begin{cases} 0.5\left(\mathbf{h}_{\hat{g}} - \mathbf{h}_g\right) & \text{if } \left|\mathbf{h}_{\hat{g}} - \mathbf{h}_g\right| \leqslant 1, \\ \left|\mathbf{h}_{\hat{g}} - \mathbf{h}_g\right| - 0.5 & \text{otherwise.} \end{cases}$$

The projected features are then combined, by concatenation, with the visual features emitted directly by ViT.

5. Experiment

To evaluate the effectiveness of the proposed multi-task multimodal classification method, some experiments were performed: the experimental setting and the results obtained are described and discussed below. The experiments were performed on a Desktop PC with an Intel i7-10700k CPU, 64 GB of RAM, and an NVIDIA RTX 3080 GPU. All models were implemented in Python using the popular PyTorch library.

5.1. Setting

We compared the proposed model with the following three baselines:

- ViT, as the backbone of our method: this is a version of ViT pre-trained on ImageNet and fine-tuned on our image data. This model uses only visual features. Similarly, we also experimented with ResNet50 [36];
- *ContextNet*: this is the method proposed by Garcia et al. [26]. The model is based on ResNet50 and uses node2vec to help the CNN learn context-aware features, even if these are not used as an additional input modality during training. For a fair comparison, we replicated *ContextNet* and trained it on our KG rather than the original KG based on SemArt;
- Our previous preliminary method [11]: the previous version of our current method, in which the visual embeddings, obtained with ResNet, are projected into the multidimensional feature space of graph embeddings obtained with node2vec and are concatenated before the output layer. It is worth noting that a smaller KG was used in our previous work; therefore, for a fair comparison, the method was retrained.

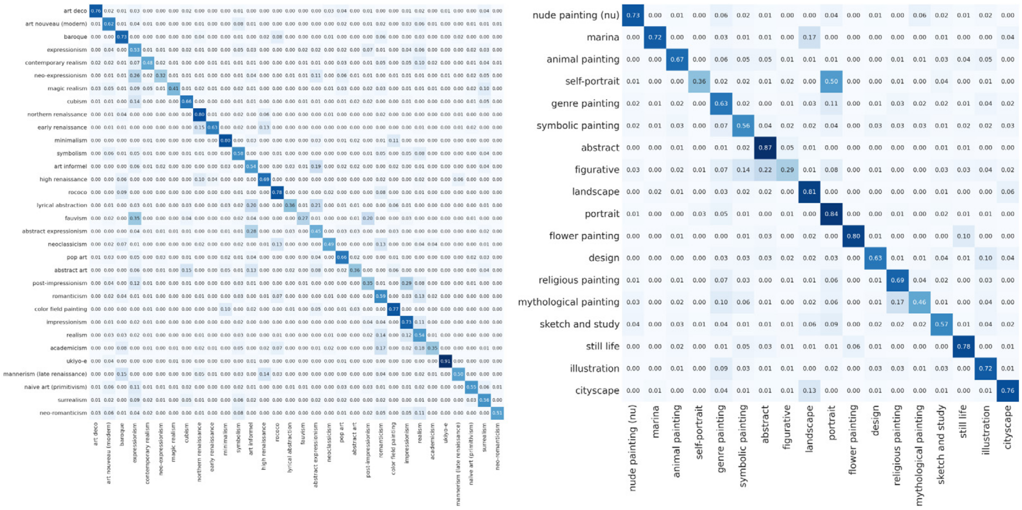
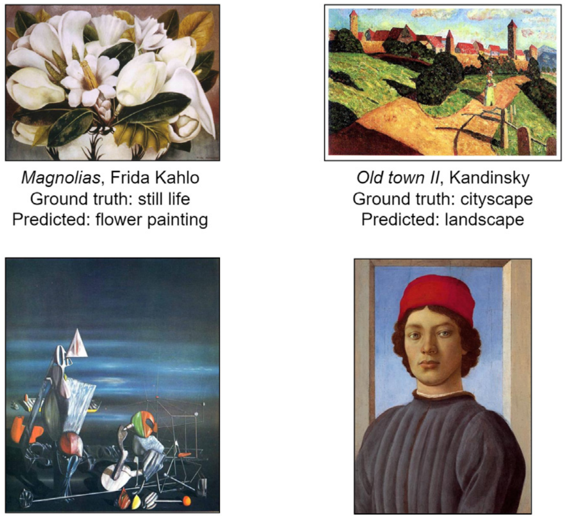


Fig. 4. Confusion matrices obtained with the proposed method for the style (left) and genre (right) classification task.



Magnolias, Frida Kahlo
Ground truth: still life
Predicted: flower painting

Old town II, Kandinsky
Ground truth: cityscape
Predicted: landscape

Slowely toward the North, Tanguy
Ground truth: landscape
Predicted: symbolic painting

Portrait of a young man with red cap, Botticelli
Ground truth: religious painting
Predicted: portrait

Fig. 5. Examples of misclassifications made with the proposed method.

We have not compared our method with recent works using Graph Neural Networks because, as previously mentioned, they adopt a transductive rather than an inductive approach.

For all models, the artwork images were resized to 224 × 224, and normalized using the mean and standard deviation of ImageNet. As an optimizer, we used Adam with a learning rate of 3×10^{-5} and a batch size of 32; in addition, a dropout with dropout rate of 0.4 was applied to the final hidden layer. As for the training of GAT in our proposed method, a dropout with dropout rate of 0.4 was applied to the hidden layer and the Adam optimization algorithm was used with a learning rate of 0.01.

It is worth noting that graph embeddings should not be learned on the entire graph, otherwise a bias would be introduced so that the model has already seen the test entities and their connections with the rest of the graph. In fact, the context information learned during training has already served to allow the model to generalize beyond just the visual features and so we assume that at test time only the visual appearance of the artwork is known to the model. To this aim, we randomly divided our graph (and consequently the image set) into three sets: 70% for training, 15% for validation, and 15% for testing. The splitting was done in a *stratified* way, preserving the percentage of artworks for each genre and style class. The validation set was used to tweak the hyper-parameters and to implement an early stopping criterion. Embeddings were only learned from the "training" graph. It is worth pointing out that while this does not happen with our data because even rare classes are well represented, the weighted loss we used to mitigate the class imbalance would result in biased results if multiple single-membered classes exist. Therefore, we warm the reader to use caution when adopting such a method.

For each experiment, we measured the top-1 and top-2 accuracy, which computes the number of times the correct class was among the top-2 predicted classes, and the macro F1-score, which calculates the F1-score for each class and then their average.

5.2. Results

Tables 2 and 3 show the results obtained by the compared methods for the single-task and multi-task learning, respectively. The tables also show the results obtained with ResNet, alone or in place of ViT as the backbone for extracting visual features in the proposed method. A first observation that can be drawn is that ViT is generally preferable to ResNet. This is true when they are considered alone, but it is also reflected in the results obtained with the method proposed in this work, when one or the other is used as a backbone.

Another consideration is that the proposed method, which uses a GAT to extract the context features, generally shows superior performance to all other methods both when using ResNet or ViT. In particular, the best results over all metrics are obtained with ViT. This confirms, as already noted in [26], that the joint exploitation of visual and contextual features improves performance compared to using visual features alone. This is not always true when considering *ContextNet* or our previous preliminary work, which use node2vec, but this was expected. The GAT-based approach, in fact, takes advantage of the heterogeneity of *ArtGraph*, while node2vec neglects this property of the graph.

As a third observation, we can see from the results that multi-task learning is generally preferable to single-task learning, i.e. optimizing the two tasks simultaneously is better than optimizing them independently of each other. This confirms the findings already reported in [8]. This superiority is not exhibited by all methods for all metrics; however, the best overall results for style and genre classification were obtained with the proposed method when they were optimized simultaneously.

Finally, one aspect that makes the task of predicting the attributes of an artwork challenging is the fact that it is very difficult to distinguish artworks with very similar genres or styles. This is confirmed by the confusion matrices obtained by the proposed method when used in multi-task mode, shown in Fig. 4. As can be seen, the model often confuses self-portraits with portraits, mythological paintings with religious paintings, or neo-expressionism with expressionism. As found in other works, such as [7], this is a well-known difficulty in this domain. A qualitative analysis of the misclassifications, in fact, as shown in Fig. 5, reveals that the model makes mistakes that would be difficult to classify even for humans. Another proof of the difficulties encountered with ambiguous class memberships is finally given by the results shown by the top-2 accuracy, which is significantly higher in all cases than the top-1 accuracy.

6. Conclusion

In this paper, we have addressed the problem of automatic art analysis by proposing a method that jointly exploits visual and Knowledge Graph embeddings with the aim of improving performance compared to using only visual features. The method leverages the increasingly popular paradigms of Graph Neural Networks and Vision Transformers and, contrary to previous methods, it works in a completely inductive way. This allows the model to work even in real-time when the only information available on a novel artwork is only its visual aspect.

To achieve this, in this paper we have also presented *ArtGraph*, an artistic Knowledge Graph, which can provide art historians with a rich and easy-to-use tool to perform fine art analysis. An art historian, in fact, rarely analyzes artworks as isolated creations, but typically studies how different paintings, even from different periods, relate to each other, how artists from different countries and/or periods have exercised a influence on their works, how artworks completed in one place migrated to other places, and so on. This effort can foster the dialogue between computer scientists and humanists that is currently sometimes lacking [37]. In addition, we believe that *ArtGraph*, which we have made publicly available, will provide the Pattern Recognition and Computer Vision community with a good basis for further research on automatic art analysis.

As future work, we want to tackle other significant tasks, such as multi-modal retrieval [38] and artwork captioning [39]. In addition, we observe that *ArtGraph* can be extended in several ways. First of all, it can be easily extended with new data relating to both public and private data. More data means more knowledge is encoded in the graph and this can improve the generalizability of predictive models. Secondly, the proposed multi-modal classification method is intrinsically extensible as new input modalities can be used and concatenated to existing ones to provide the model with different perspectives on the same input. For example, a natural extension is the use of word embeddings obtained from textual descriptions of the artworks. Finally, the model can be used for *link prediction* in order to infer new relations and thus build a more complete Knowledge Graph. This strategy could allow *ArtGraph* to evolve over time, including contemporary art, and suggests its use in semi-supervised learning settings [40].

CRediT authorship contribution statement

Giovanna Castellano: Conceptualization, Formal analysis, Investigation, Methodology, Resources, Validation, Writing – original draft, Writing – review & editing. **Vincenzo Digeno:** Data curation, Formal analysis, Investigation, Methodology, Software, Validation, Visualization, Writing – review & editing. **Giovanni Sansaro:** Data curation, Formal analysis, Investigation, Methodology, Software, Validation, Visualization, Writing – review & editing. **Gennaro Vessio:** Conceptualization, Formal analysis, Investigation, Methodology, Project administration, Resources, Software, Supervision, Validation, Visualization, Writing – original draft, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

G. V. acknowledges funding support from the Italian Ministry of University and Research through the PON AIM 1852414 project.

References

[1] A. Hogan, E. Blomqvist, M. Cochez, C. d'Amato, G.D. Melo, C. Gutierrez, S. Kirrane, J.E.L. Gayo, R. Navigli, S. Neumaier, et al., Knowledge graphs, ACM Comput. Surv. 54 (4) (2021) 1–37.

[2] X. Wang, H. Wu, C.-H. Hsu, Mashup-oriented api recommendation via random walk on knowledge graph, IEEE Access 7 (2018) 7651–7662.

[3] D. Calvanese, B. Cogrel, S. Komla-Ebri, R. Kontchakov, D. Lanti, M. Rezk, M. Rodriguez-Muro, G. Xiao, Ontop: Answering SPARQL queries over relational databases, Semant. Web 8 (3) (2017) 471–487.

[4] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, Nature 521 (7553) (2015) 436–444.

[5] G. Castellano, G. Vessio, Deep learning approaches to pattern extraction and recognition in paintings and drawings: An overview, Neural Comput. Appl. (2021) 1–20.

[6] E. Cetinic, T. Lipic, S. Grgic, Fine-tuning convolutional neural networks for fine art classification, Expert Syst. Appl. 114 (2018) 107–118.

[7] C. Sandoval, E. Pirogova, M. Lech, Two-stage deep learning approach to the classification of fine-art paintings, IEEE Access 7 (2019) 41770–41781.

[8] G. Strezoski, M. Worring, Omniart: A large-scale artistic benchmark, ACM Trans. Multimed. Comput. Commun. Appl. (TOMM) 14 (4) (2018) 1–21.

[9] M. Cornia, M. Stefanini, L. Baraldi, M. Corsini, R. Cucchiara, Explaining digital humanities by aligning images and textual descriptions, Pattern Recognit. Lett. 129 (2020) 166–172.

[10] N. Garcia, G. Vogiatzis, How to read paintings: semantic art understanding with multi-modal retrieval, in: Proceedings of the European Conference on Computer Vision (ECCV) Workshops, 2018.

[11] G. Castellano, G. Sansaro, G. Vessio, Integrating contextual knowledge to visual features for fine art classification, in: Workshop on Deep Learning for Knowledge Graphs (DL4KG 2021), 2021.

[12] R.S. Arora, A. Elgammal, Towards automated classification of fine-art painting style: A comparative study, in: ICPR, 2012, pp. 3541–3544.

[13] G. Carneiro, N.P. da Silva, A. Del Bue, J.P. Costeira, Artistic image classification: An analysis on the PRINTART database, in: ECCV, 2012, pp. 143–157.

[14] F.S. Khan, S. Beigpour, J. Van de Weijer, M. Felsberg, Painting-91: A large scale database for computational painting categorization, Mach. Vis. Appl. 25 (6) (2014) 1385–1397.

[15] E. Cetinic, T. Lipic, S. Grgic, A deep learning perspective on beauty, sentiment, and remembrance of art, IEEE Access 7 (2019) 73694–73710.

[16] B. Saleh, K. Abe, R.S. Arora, A. Elgammal, Toward automated discovery of artistic influence, Multimedia Tools Appl. 75 (7) (2016) 3565–3591.

[17] S. Karayev, M. Trentacoste, H. Han, A. Agarwala, T. Darrell, A. Hertzmann, H. Winnemoeller, Recognizing image style, in: BMVC, 2014.

[18] L. Chen, J. Yang, Recognizing the style of visual arts via adaptive cross-layer correlation, in: ACM MM, 2019, pp. 2459–2467.

[19] N. Van Noord, E. Hendriks, E. Postma, Toward discovery of the artist's style: Learning to recognize artists by their artworks, IEEE Signal Process. Mag. 32 (4) (2015) 46–54.

[20] G. Castellano, E. Lella, G. Vessio, Visual link retrieval and knowledge discovery in painting datasets, Multimedia Tools Appl. 80 (5) (2021) 6599–6616.

[21] B. Seguin, C. Striolo, F. Kaplan, et al., Visual link retrieval in a database of paintings, in: ECCV, Springer, 2016, pp. 753–767.

[22] E.J. Crowley, A. Zisserman, The art of detection, in: ECCV, 2016, pp. 721–737.

[23] N. Gonthier, Y. Gousseau, S. Ladjal, O. Bonfait, Weakly supervised object detection in artworks, in: ECCV, 2018.

[24] P. Hall, H. Cai, Q. Wu, T. Corradi, Cross-depiction problem: Recognition and synthesis of photographs and artwork, Comput. Vis. Media 1 (2) (2015) 91–103.

[25] X. Shen, A.A. Efros, A. Mathieu, Discovering visual patterns in art collections with spatially-consistent feature learning, ICPR (2019).

[26] N. Garcia, B. Renoust, Y. Nakashima, Contextnet: Representation and exploration for painting classification and retrieval in context, Int. J. Multimed. Inf. Retr. 9 (1) (2020) 17–30.

[27] A. Grover, J. Leskovec, Node2vec: Scalable feature learning for networks, in: ACM SIGKDD, 2016, pp. 855–864.

[28] W. Zhao, D. Zhou, X. Qiu, W. Jiang, How to represent paintings: A painting classification using artistic comments, Sensors 21 (6) (2021) 1940.

[29] A. Efthymiou, S. Rudinac, M. Kackovic, M. Worring, N. Wijnberg, Graph neural networks for knowledge enhanced visual representation of paintings, 2021, arXiv preprint arXiv:2105.08190.

[30] C.B.E. Vaigh, N. Garcia, B. Renoust, C. Chu, Y. Nakashima, H. Nagahara, GC-NBoost: Artwork classification by label propagation through a knowledge graph, 2021, arXiv preprint arXiv:2105.11852.

[31] S. Zhang, H. Tong, J. Xu, R. Maciejewski, Graph convolutional networks: A comprehensive review, Comput. Soc. Netw. 6 (1) (2019) 1–23.

[32] T. Baltrušaitis, C. Ahuja, L.-P. Morency, Multimodal machine learning: A survey and taxonomy, IEEE Trans. Pattern Anal. Mach. Intell. 41 (2) (2018) 423–443.

[33] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al., An image is worth 16x16 words: Transformers for image recognition at scale, 2020, arXiv preprint arXiv:2010.11929.

[34] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, Y. Bengio, Graph attention networks, 2017, arXiv preprint arXiv:1710.10903.

[35] S. Ruder, An overview of multi-task learning in deep neural networks, 2017, arXiv preprint arXiv:1706.05098.

[36] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.

[37] G. Mercuriali, Digital art history and the computational imagination, Int. J. Digital Art Hist.: Issue 3, 2018: Digital Space Archit. 3 (2019) 141.

[38] N. Jain, C. Bartz, T. Bredow, E. Metzenthin, J. Otholt, R. Krestel, Semantic analysis of cultural heritage data: Aligning paintings and descriptions in art-historic collections, in: International Conference on Pattern Recognition, Springer, 2021, pp. 517–530.

[39] E. Cetinic, Towards generating and evaluating iconographic image captions of artworks, J. Imag. 7 (8) (2021) 123.

[40] J.E. Van Engelen, H.H. Hoos, A survey on semi-supervised learning, Mach. Learn. 109 (2) (2020) 373–440.