

Computational Analysis of Content in Fine Art Paintings

Diana Kim

Computer Science
Rutgers University
New Brunswick, NJ, U.S.A
dsk101@rutgers.edu

Jason Xu

Operations Research and Financial Engineering
Princeton University
Princeton, NJ, U.S.A
jasonx@princeton.edu

Ahmed Elgammal

Computer Science
Rutgers University
New Brunswick, NJ, U.S.A
elgammal@cs.rutgers.edu

Marian Mazzone

Art and Architectural History
College of Charleston
Charleston, SC, U.S.A
marian.mazzone@gmail.com

Abstract

We propose a deep learning algorithm that can detect content and discover co-occurring patterns of the content in fine art paintings. The following intellectual merits are the motivations of our project.

First, the content detection provides a baseline of Computational Iconography (CI), which is to understand what objects/subjects can be seen in fine art paintings. Second, we argue that the found co-occurring patterns chart meaningful connectivity across content in art. Third, we imbed our system in Computational Creativity (CC) in a broad sense. By the nature of our system of machine learning, it creates informative connections between different modalities (images/words), which are not initially constructed or intentionally specified. Our system is automatically trained to discover the connective patterns reflecting artists' creativity, which are latent in the large dataset of paintings.

To build a content detector, we adopted an Inception-V3 (ImageNet) and fine-tuned it over 40,000 paintings with the words extracted from their titles. We validate that our system detects content information fairly (68% precision rate at the top content). Also, we find that the last fully connected layer parameters of Inception-V3 are trained to encode general co-occurring patterns between content. We validate that the co-occurrence can be interpreted as relatedness among content in art.

Introduction

In this paper, we present a computational method that can understand the content of fine art paintings. By bringing our problem on the broader stance of general art, we highlight our system interprets art, especially in terms of the content, which is one of the three principles for understanding art: form, content, and context (Dyke 1887; Lowry 1967).

More specifically, we adopt a deep learning approach and argue how it automatically creates many virtual connections from a target painting to the multiple relevant pieces of information called content: the objects, activity, or other information that can be seen in the painting. First, we implement a content detector to connect a fine art painting (image) and relevant output words (content). It creates plenty of textual information about a given visual entity. Second, as a by-product of the content detector, we find that distributed vector representations, of mutual distances capture the general

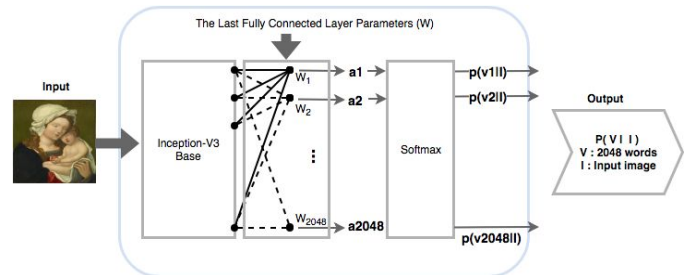


Figure 1: Content Detector

co-occurrence patterns among content. We prove that the co-occurrence of content can be interpreted as the relevance among content in art, which is embedded in large training set of paintings.

Our proposal for the computational content analysis can have the following practical and intellectual merits. The found virtual connections would be useful to associate words to the images/words we are focused on, so it drives us to other resources flowing into other relevant paintings. It enables us to reach more paintings from a few key words of general content. This suggests a feasible application of our system to improve general accessibility to digital art retrieval systems. Currently, art retrieval systems require highly specialized knowledge such as title, author, genre, time period, or style of paintings, which ordinary users may not know well.

Furthermore, building computational models for understanding human creative products can be a fundamental part of the field of Computational Creativity (CC). We argue our computational model links to broad perspectives of CC. Although our methodology does not precisely articulate how artists' mind operates on their creative artifacts or create novel products, it does focus on artworks which are objects of human creation, and it may give us insight into the pattern of connections among concepts, words, and visuals that artists use when making their images.

Our machine takes in many paintings as inputs and learns to connect images and words as a reflection of how the input artifacts are presenting. The connections are not pre-constructed or designed by the authors or from any external knowledge of art. They are instead solely the result of the

huge processing capacity of the machine to work with images and words. We believe that by analyzing such a large number of paintings, the machine is able to reflect the associative patterns of images and co-occurring concepts that human artists may be using when they create their artworks. Our system may be related to the broad definition of CC (Bown 2012), in that a computationally creative system is not necessarily modeled on the human mind or on human goals, but does apply to the occurrence of creation.

In previous computational art analysis, most research works have focused on visual appearance and its descriptions, i.e., visual forms, such as brush strokes (Elgammal, Kang, and Den Leeuw 2018; Hendriks and Hughes 2009) and stylistic analysis (Kim et al. 2018; Elgammal et al. 2018). However, as we consider three art principles, which are primary elements for understanding art (Lowry 1967), analysis grounded purely in visual forms may not be a sufficient approach. We can better appreciate art if we understand content, including the subject matter and historical context of interpreting that content. In art history, this approach is called the study of iconography and iconology, with its most notable practitioner being Erwin Panofsky (1892–1968). Hence, we devise a computational framework for content and it provides the baseline work for Computational Iconography (CI).

To build our content detector, we adopt and fine-tune a deep neural network architecture, Inception-V3 (Szegedy et al. 2016), for which the input is an image (painting) and the output is a probability mass function (pmf) as shown in Figure 1. In the model, the pmf’s support domain V is 2,048 words, so through probabilistic representation, we can quantify the relevance of each word to an input image I . For fine-tuning, we only re-train parameters of the last fully connected (FC) layer and other parameters are transferred from a pre-trained ImageNet. While the training proceeds, we observe an interesting property: the network starts learning to capture associative patterns between the output words. For more details, (W_i) in Figure 1, the weight vectors are trained to be a distributed representation set, i.e., their mutual distances can encode certain relationships between the content (words) in paintings. Although we intentionally train the machine to create linkages between an image and words, but the machine also autonomously learns to capture relationships among words, too.

We can observe the following features.

- Words denote concepts that are visually similar from the perspective of the machine, if (and only if) the word representations are likely to be close each other.
- Concepts often co-occur within a painting, if (and only if) the corresponding word representations are likely to be close each other.

From the above analysis, we can notice distinct characteristics of our vector representation through differentiation with the word embedding systems in Natural Language Processing (NLP). In NLP, word embedding models (Mikolov et al. 2013) encode syntactic or semantic similarities between words through the context of the likeness to their neighboring words. On the other hand, our embedding sys-

tem encodes word relationships based on the visual similarities or co-occurring patterns of the concepts over a whole set of fine art paintings, i.e., instead of adjacent words, paintings become major contextual resources to extract relationship between words.

We specially point out that the second co-occurrence property could offer us clues about which subjects and concepts typically occur simultaneously in paintings. Using this, we could easily find repeated motives across paintings. In practice, it is one of main tasks of iconography, but may not be easy if we only have the pure semantics of the words available, or if we look for them only through the human naked eyes. For instance, in our distance analysis, the two words ‘Virgin’ and ‘Angel’ are the closest words. Their pure semantics are not highly correlated, but we confirm that they are two primary co-occurring components for the subject of ‘Annunciation’. It is a popular theme in paintings during the Middle Ages and Renaissance.

The method may not be able to find immense and delicate symbolic meanings as art historians have done, such as Erwin Panofsky’s discovery of a connection between lilies and Mary as a symbol of her chastity in *Mérod Altarpiece* (Panofsky 1971). However, clues are sometimes enough to initiate deeper directed studies, especially when we deal with the massive archive of paintings. Furthermore, we also know a fact: iconographic analysis should begin with the object that can be seen from the art works (Munsterberg 2009).

In summary, we claim the followings.

1. Our system detects content information fairly well. As the system is designed to detect multiple labels, the loss objective in training does not measure actual performance well. We validate the performance through the following alternative methods: (1) Comparison between machine pmfs and words populations. (2) Human subjects survey with students in art history.
2. Our system discovers co-occurring patterns and it implies certain relatedness among content in art. We validate the claim through the correlation analysis between the degree of co-occurrence (mutual distances between the vector representations for two key words) and the relevance (number of results to searching queries of intersecting two key words).

In the following sections, we will explain the whole procedure of implementing a content detector and achieving distributed representations. We will also explain evaluation procedures and its results. In the last discussion section, we will draw a practical application of our system on current digital art searching platforms.

Related Works

Our problem shares a common goal with some prior researches about computational content analysis in art collections (Carneiro 2011; Carneiro et al. 2012; Crowley and Zisserman 2014; Picard, Gosselin, and Gaspard 2015).

To our best knowledge, Gustavo (2011) (Carneiro 2011)’s graph-based learning algorithm was the first computational approach to detect content in art works. He annotated digital art prints with 28 pre-defined semantic labels. Before

his work, most computational art analysis had focused on visual forms such as brush strokes (Polatkan et al. 2009; Hendriks and Hughes 2009) or stylistic analysis (Jafarpour et al. 2009).

Later, Gustavo *et al.* (Carneiro et al. 2012) proposed other computational approaches (random, bag of features, label propagation, and inverted label propagation) to detect 75 content classes from monotonic paintings. By dividing the targeted annotations into global semantic, local composition, and local pose, they tried to detect more structured semantics from the more general paintings than their previous works.

Elliot *et al.* (Crowley and Zisserman 2014) used a transfer learning scheme. They showed that object classifiers trained by natural images can effectively detect objects in paintings. They compared the performances of two Support Vector Machine (SVM) classifiers, in which each machine is trained with one of two feature sets: Fisher vector representations or vector collections from an intermediate layer of the Convolutional Neural Network (CNN). In the result, the CNN outperformed the Fisher vector representation. However, their experiments were limited to object-oriented concepts, such as chair, bird, and boat, and there were only 10 classes.

David *et al.* (Picard, Gosselin, and Gaspard 2015) used the same methodologies as the work of Elliot *et al.* (Crowley and Zisserman 2014), but applied them to annotate cultural heritage collections. During the experiment, they classified artifacts in one of 459 semantic classes. Differently to the result of Elliot *et al.* (Crowley and Zisserman 2014), the performance of the Fisher vector representation was slightly better than one of the CNN features.

In our methodology, we applied the deep-learning method to understand the content in fine art paintings and validated its performance. For the data set, we did not use any pre-defined words like those of previous works. Instead, we collected words from the titles of paintings and selected 2,048 words based on the words' statistics. Along with content detection, we also found associative patterns (co-occurring or visually similar) between the content in paintings.

Methodology

Our primary goal is to design a system that can represent a conditional pmf: $P(V|I)$, where V represents a word whose domain is a 2,048 words set and I is an input image. Based on probability, we will try associating highly probable words with an input image and validate their association.

Architecture

To design the probabilistic system, we utilize a multiple labeling training by modifying an original machine learning algorithm, Inception-V3. We train the same network architecture by using its original objective function. However, as the original algorithm can handle only multiple class problems (class labels are mutually exclusive) setting only one class as probability one, we have to change the framework to enable it to carry multiple non-zero probabilities. For a K multi-class problem, the network's output produces a pmf whose k -th element implies the probability of the k -th word

(v_k) given an input image I . The original objective function is a softmax cross-entropy for training data samples. Let a_k be the k -th value before the softmax layer in the network of Figure 1. Then the output probability P_k and the objective function E of N samples are the following.

$$P_k = P(V = v_k|I) = \frac{\exp^{a_k}}{\sum_k^K \exp^{a_k}} \quad (1)$$

$$E = - \sum_n^N \sum_k^K I_{k,n} \cdot \log_e(P_k)$$

In equation 1, $I_{k,n}$ is an indicator function stating whether or not the n -th sample belongs to class k . In our project, to handle the multiple labels, we re-define an objective function E' with a $J_{k,n}$ instead of the indicator function.

$$P_k = P(V = v_k|I) = \frac{\exp^{a_k}}{\sum_k^K \exp^{a_k}} \quad (2)$$

$$E' = - \sum_n^N \sum_k^K J_{k,n} \cdot \log_e(P_k)$$

In equation 2, the summation of $J_{k,n}$ over all k is equal to one ($\sum_k^K J_{k,n} = 1$), and each value is $J_{k,n} = \frac{1}{L}$, if the n -th input image has a k -th word and the total number of labels of the image is L . We can interpret the E' as a negative log likelihood function if we draw a case in which multiple words for each sample image are independently generated by the P_k . Suppose we have three labels (v_1, v_2, v_3) for an image, then a $P(v_1, v_2, v_3|I)$ equals $\prod_{k=1}^3 (P_k)$. Then we can compute the $E' = -\frac{1}{3} \sum_{k=1}^3 \log_e(P_k)$ for the sample image and its labels. If we consider each pair (I, v_1) , (I, v_2) , (I, v_3) as independent samples, then it is the same as the original multiple class objective function E except for the normalization factor of $\frac{1}{3}$. Internally, we use multi-class training L times and compensate its multiple uses by dividing it by L . In this sense, the modification does not harm the primary concept of cross entropy that the original algorithm intends and it can handle multiple label training.

The Last Fully Connected Layer Weights

In NLP, a skip-gram model (Mikolov et al. 2013) can learn distributed representations of words by capturing statistical patterns with their neighboring words in a text corpus. If two words' neighboring words are often similar, their representation also become close. Inspired by the idea, we hypothesize that the last layer weights of our network can also encode an associative relationship between the 2,048 output words. If two images are visually similar but labeled by two different words, then the two word representations are expected to be close.

We can think of two cases in which images are the same or visually similar, but labeled by different words. For the first case, in general, low-level concepts are visually similar if they have a common ancestor in the concepts' hierarchical system. For example, specific kinds of flowers such as lanaculus, rose, or camellia are necessarily mapped into very

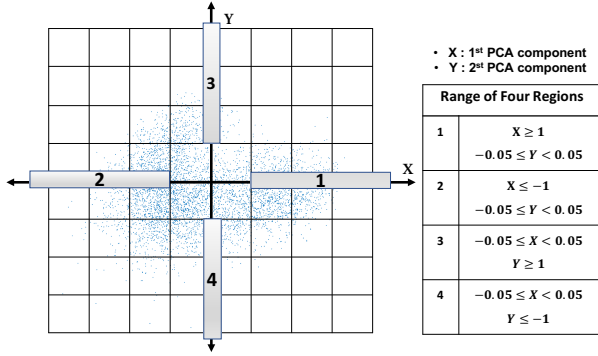


Figure 2: Four Regions in two components of PCA space (Blue dots: 2-D PCA transformed points of validation data)

close points in the top hidden layer of a neural network layer, but each of the points are to be labeled with different names.

For the second case, if some concepts often co-occur in paintings, it corresponds to the case in which the painting images are the same but labeled by different concepts. For instance, Christ, cross, angel, and a subject of lamentation are often delineated in a painting. Similarly, Madonna, child, and Saints often co-occur, too.

In this context, we examine the last weights W (2,048 \times 2,048) in Figure 1 as distributed representations for the 2,048 concepts, and confirm that they are close to one another if (and only if) their corresponding concepts are visually similar or frequently co-occurring in paintings.

Validation Methods

In this project, we have two claims. One is we can build a probabilistic system that can have higher probabilities on words more relevant to an input painting. The second one is that parameters (words distributed representations) collected from the last layer of the system can encode the relationships between the words in fine art paintings. In the following sections, we will explain how we have validated the claims.

Content Detection To validate the first claim, we conducted a survey to determine how many subjects agree with the machine’s 10 most probable words as relevant concepts. More detailed survey results and its steps are presented in the later survey section in the Experimental Result. As the second evaluation method, we performed the following experiment: we collected image embeddings from the hidden layer right after Inception-v3 base (in Figure 1) by inputting training images. Then, we learned two Principal Components Analysis (PCA) components (occupying 10% of the total variance of the embeddings). In the found PCA space, images near the points of $[c, 0]$, $[-c, 0]$, $[0, c]$, and $[0, -c]$ (in our experiment, c is 4) showed a certain degree of consistency in their content, so we set the following hypothesis and validated it.

If a machine can detect content from an input painting well, then the following two statistics will be similar to each other. One is the sample frequency histograms of

Table 1: Ranking Ranges for Each Group

Group	Ranking Ranges
Group 0	One word ranked 1 (the word itself)
Group 1	Ten words ranked 2-11
Group 2	Ten words ranked 1025-1034
Group 3	Ten words ranked 2039-2048

title-words of four groups of images, located near the points $[c, 0]$, $[-c, 0]$, $[0, c]$, and $[0, -c]$. Each group of images are the validation images that are PCA transformed to the regions defined by the ranges of 1, 2, 3, and 4 in Figure 2. Another statistic is the machine’s output pmfs, as we individually pass four simulated image embeddings into the network after the Inception-V3 base (in Figure 1). Each of the four simulated image embeddings has been computed by an approximate inverse PCA on the vectors $[c, 0]$, $[-c, 0]$, $[0, c]$, and $[0, -c]$.

Let d be the number of PCA components, s the number of validation samples, H the collected sample embeddings from the hidden layer, and g the size of the dimension of the hidden layer. As we do whitening PCA on the hidden layer embedding H ($g \times s$), a PCA transformed T ($d \times s$) can be written as

$$T = \Lambda^{-\frac{1}{2}} \cdot \Theta^T(H - m) \quad (3)$$

where m is a mean vector computed from H , Θ is a matrix ($g \times d$), whose columns are orthonormal vectors to define the PCA’s principal axes, and Λ ($d \times d$) is a diagonal matrix defining the PCA variances. By using 3, we can simulate the four embeddings \hat{h}_z ($g \times 1$) that equal $\Theta \cdot \Lambda^{\frac{1}{2}} \cdot t_z + m$, where $t_1 = [c, 0]^t$, $t_2 = [-c, 0]^t$, $t_3 = [0, c]^t$, and $t_4 = [0, -c]^t$, z is in $[1, 4]$. Now, we can compute the machine’s output distribution \hat{y}_z in 4 and the W' has the same columns of W except for the last bias column vector w_{bias} . The \hat{y}_z is the network outcome when inputting the simulated embedding \hat{h}_z into the last FC layer in Figure 1.

$$\hat{y}_z = \text{softmax}(W' \cdot \hat{h}_z + w_{bias}) \quad (4)$$

Words Distributed Representation After finishing training, we collected the last layer parameter W in Figure 1, and regarded each of the i -th rows (w_i) as the distributed representation of the i -th word. We computed cosine similarities between the representations and formed a matrix M in 5. Each component $M_{i,j}$ represents the cosine similarity between the representations of the i -th and the j -th word.

$$M_{i,j} = \frac{w_i^t \cdot w_j}{|w_i| \cdot |w_j|} \quad \forall i, j \in [1, 2048] \quad (5)$$

To find relationships between distance, we sorted each row of M in descending order and for each row, we set the first word as group 0 and collected the other three groups of words according to their rankings as shown in Table 1.

To verify that closer words in the distributed representations are more correlated words in art, we tried searching artworks in GoogleArt&Culture

(<https://artsandculture.google.com>) with queries of intersecting two words. Its first word is the group 0 word and another word is from group 1, group 2, or group 3. We posit that returning more results as we query an intersection of two words is a reflection of more connections between the words in the art domain. GoogleArt&Culture searches artworks by intersecting all input words and matching them to words in the documents in its database, which have basic information (author, title, and year) or general descriptions about paintings. Hence, the number of retrieved art works should generally decrease with successive groups 1, 2, and 3 if the distributed representations can encode correlations between words within art.

Experiment Results

Data set – Paintings and words from titles

We used a public collection of fine art paintings, the WikiArt (<https://www.wikiart.org>) data set. The collection has more than 60,000 paintings covering the Renaissance to the Modern period. Instead of using all of them, we utilized paintings drawn before the 20th-century (50,160 images) and split them into ‘Train’ (85%), ‘Validation’ (10%), and ‘Test’ (5%). We used ‘Train’ in training the Inception-V3 and defining PCA’s principal axes, ‘Validation’ for evaluations and a survey, and ‘Test’ for presenting test results.

To prepare training samples, we labeled the paintings with words from each painting’s title. All words from the titles are good sources for understanding the content of target paintings, but we do not want to use words that appear too sparsely or refer to specific entities, such as the name of a area or a person. Using the Natural Language Toolkit (NLTK) library (version 3.2.5), we removed any digits, ‘CC’ (coordinating conjunction), ‘DT’ (determiner), ‘TO’ (TO-infinitive), and ‘IN’ (preposition), and two-letter words from the titles, and labeled the paintings with the remaining 2,048 most frequent words.

All training images have at least one label. If one does not have a label, it is not used as a training sample. Many painting titles can provide informative resources to answer basic questions about the subject matter, such as what, where, who, or when (Gombrich 1985), but during some periods not all titles correlate to content in a helpful way. For example, several titles of Paul Klee (1879-1940) and Joan Miró (1893-1983)’s works refer to literary works, and many other titles in modern art are simply descriptive of shapes or colors, composed of numbers, or images are left untitled. For these reasons, in this project we use the paintings of Renaissance, Baroque, Rococo, Romanticism, Impressionism, Post-Impressionism, and Realism styles.

Fine-Tuning Inception-V3

After modifying the objective of an official model, Inception-V3 (TF-slim in Tensorflow Ver 1.4), we fine-tuned it for 300,000 steps. We only updated the last FC layer, ‘Logits’ and ‘AuxLogits’ (Szegedy et al. 2016), and other parameters were transferred from a pre-trained model. The average loss E' defined in 2 converged from an initial value of $-\log_e(\frac{1}{2048}) = 0.00048$ to a value of about

Table 2: Common Words

PCA region	common words
First and Positive	landscape, river, bridge, path, trees, forest
First and Negative	portrait, child, virgin, Madonna, man, Christ, self, young
Second and Positive	portrait, man, woman, self, young, artist, lady
Second and Negative	landscape, life, trees, beach, scene, winter, bridge

$-\log_e(0.0025)$, but it did not get lower.

The main cause of our high converged error rate is the intrinsic property of titles in artworks. Basically, titles can have various words choices, and even in subject similar paintings, depending on author’s focal points, we can choose words that are semantically different. In other words, there is not only one correct title for an image. Hence, our simple probabilistic output modeling, $P(V|I)$, conditioning only on an input image, may not be sufficient to capture the variance of titles.

Evaluations

We validated our three claims by using the three methodologies described in the Validation Methods section of the Methodology. In the following three subsections, we present the results of the evaluations.

Content Detector: (1) comparison between machine pmfs and words populations We compared two statistics. The first statistic is the machine output pmf as inputting a simulated image embedding. Four simulated embeddings were computed by conducting inverse PCA approximations on the four vectors: $[4, 0]$, $[-4, 0]$, $[0, 4]$, $[0, -4]$ and from them we gained four pmfs. The pmfs’ 15 top ranked words are presented in four left-handed figures (blue) in Figure 3.

The second statistic is a relative frequency of each title word in a group of images. Four groups of images are collected from the ranges defined in Figure 2. The top ranked 15 words of each group are presented in four right-handed figures (red) in Figure 3.

To consider their similarity, in each row, we compared the left and right figures. Then, we listed the common words in Table 2. We observed that at least six words were common and were semantically aligned with one another, even when they were not perfectly matched. It is natural for them not to be exactly the same each other because the results of the first column are approximately simulated from the first two PCA components, and do not consider all dimensions. Interestingly in Figure 3, there were two considerable concepts: landscape (1st and 4th row) and portrait (2nd and 3rd row). One possible explanation for the result may be the dominant majority of the portrait and landscapes in our data set. In the WikiArt data set, there were 18 different genres, but 37% of the samples were the two genres.

Content Detector: (2) survey results We conducted a survey to evaluate how the machine’s highly probable words were relevant to an input image. In the survey, we randomly selected 40 validation images and annotated them with the 10 most probable words based on the machine output pmf. It was a blind test and required subjects to do the following (to quote): “Please check all the words that can describe each

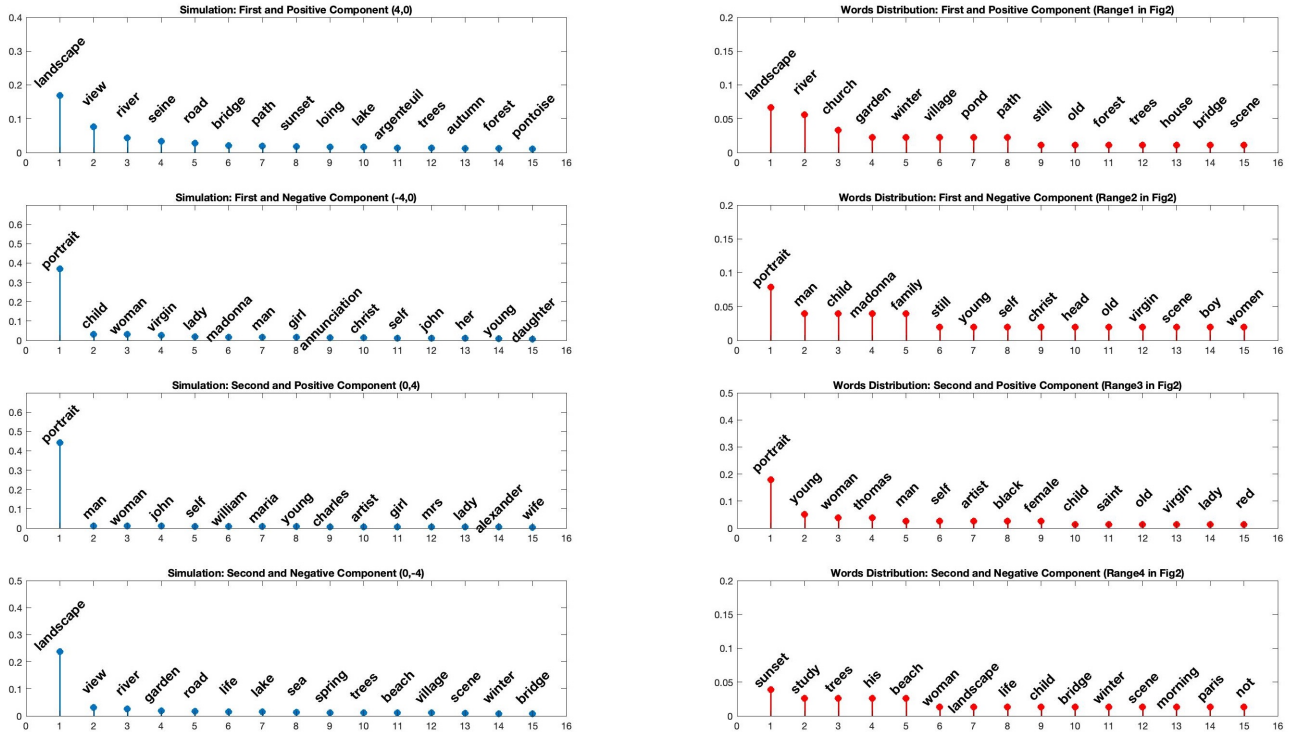


Figure 3: Comparison of two statistics: word detection and word populations within four pre-defined PCA regions

Table 3: Ten Survey Results

Title	Image	P	Machine Annotation (Precision Q = 3 Y = 5)
balchik		1.0	rock(0.8), sea(0.9), cliff(0.9)
the conversation		1.0	portrait(0.5), woman(0.9), girl(0.6)
portrait of old woman		1.0	portrait(0.9), head(0.8), woman(0.9)
country boy		0.67	portrait(0.85), seated(0.85), death(0.15)
annunciation		0.67	virgin(0.9), annunciation(0.85), saint(0.46)
a portrait of a christian de falbe		0.33	dog(0.9), woman(0.0), dancer(0.1)
venice		0.33	paix(0.0), house(0.9), bridge(0.1)
cristo no horto		0.0	portrait(0.0), child(0.1), virgin(0.4)
the decline of the Carthaginian empire		0.0	night(0.1), interior(0.0), tavern(0.0)
allegory of air		0.0	jerome(0.1), portrait(0.0), dancing(0.0)

painting. Do not check any words if none are relevant.” At least 12 graduate students in art history responded for each of the survey images.

We set thresholds from 0 to 1 with a step size of 0.1, and obtained a correct word set based on the levels. For example, for a threshold of 0.3, we considered words as right answers only when more than three out of 10 people agreed with

the word. Let the Q denote the number of top words and $q_u(Y)$ the number of correct words at threshold Y . Then, a precision@ Q at threshold Y over the $U = 40$ images can be defined as

$$P@Q(Y) = \frac{1}{U} \sum_u \frac{q_u(Y)}{Q} \quad (6)$$

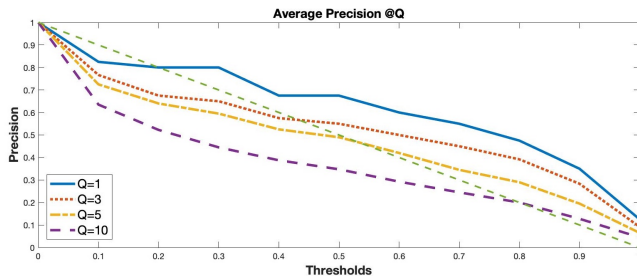




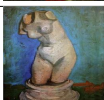


Figure 4: Survey Average Precision Rates $P@Q = 1, 3, 5,$ and 10

Table 4: Author Selected Test Result

Title	Image	Machine 10 most probable words
the bouquet		portrait, rose, woman, girl, lady flowers, young, red, roses, miss
garden in bloom		apple, trees, oak, orchard, blossom park, bloom, tree, landscape, grove
villa torlonia fountain		view, bridge, evening, park, landscape garden, fountain, pond, street, gardens
mary with child		child, madonna, portrait, girl, virgin woman, young, lady, peasant, maria
plaster statuette of a female torso		torso, blue, still, woman, life, portrait, jug, study, nude, plaster

In Figure 4, we present the precision results for $Q=1, 3, 5,$ and 10 . As Q increases, the precision values decrease and when the threshold is 0.5, the precision values are 0.68, 0.55, 0.5, and 0.35 at $Q=1, 3, 5,$ and 10 . This result validates the performance of our content detector in two senses. First, the most probable word shows a 68% average precision rate as we set the right words only when more than half of subjects agree on them. Second, as the Q increases, the corresponding precision rate drops. It implies that the machine's less probable words do not contribute to increasing the precision rate. Hence, we can see the ranking of words in the machine pmf is correlated with the subjects' responses.

To examine the quality of our system, we listed ten survey results ($Q = 3$ and $Y = 0.5$) in order of the precision rates in Table 3 and characterized them. For the high-rated (left-hand) results, most are expressed typically and simply in terms of each genre. On the other hand, for low-rate (right-hand) examples, their main figures are expressed as relatively small in complex circumstances, or a portion of the figures has characteristics that often represent other content. For instance, the third 'cristo no horto' depicts Christ,

Table 5: Number of GoogleArt&Culture Search Results

Group	Group 1	Group 2	Group 3
Averaged number of results	12,805	4,983	3,554

Table 6: Descending ordered words

Word	15 relevant words
tree	trees, pine, oak, olive, bloom, pines, orchard landscape, oaks, blossom, grove, willow, forest, asylum, peach
christ	cross, lamentation, angels, homo, ecce, deposition, virgin holy, adoration, saints, baptist, entombment, ancestors, jesus, crucifixion
angel	virgin, annunciation, vision, baptist, angels, penitent, tobas madonna, resurrection, magdalene, jesus, death, creation, elijah, allegory
rose	bouquet, wildflowers, flowers, roses, lilies, pink hollyhocks, violets, irises, lily, vase, Japanese, nasturtiums, iris, daisies
nude	female, seated, reclining, standing, bather, bath, naked model, back, hair, woman, torso, nudes, herself, male
lighthouse	seascape, tide, sunset, sailboats, lunar, harbor tower, marseille, moonrise, calm, coast, channel, maggiore, steppe, newport
virgin	madonna, child, assumption, holy, coronation, saints angels, annunciation, adoration, christ, mary, trinity, birth, enthroned, baptist

but he wears a mantle of blue, which often represents his mother. He may be wrongly detected as the virgin Mary. In the second example, 'venice', the rail of the window may be the reason why the machine detects the bridge as the third word. For further references, we selected five test-set results in Table 4. For each example, the 10 most probable words were annotated based on the machine's pmf outcome.

Words Distributed Representation As described in the Validation Methods in Methodology, by pairing two words (a word of group 0 and another word from one of the groups 1, 2, or 3), we searched GoogleArt&Culture and averaged the number of returned art works for each group. In the experiment, we only considered the top 400 words among the machine's 2,048 output domain words. The upper words are more frequent and account for more than 65% of the words frequencies, so we regard them as a representative set. The three groups' results were averaged over 400 words and presented in Table 5. It shows that the number of results decreases by 60% from group 1 to 2 and by 28% from group 2 to 3. Hence, based on the assumption that having more search results implies more connections, we can argue that closely distributed representations are likely to represent stronger relationships between words. We presented seven examples in Table 6. Based on distance analysis, we enumerated the 15 closest words for each example.

Discussion

Nowadays, many museum websites provide services to allow web users to search their digital collections through matching the user's words to basic text information they already have. The text description can refer to the title, author, genre, time period, style, or sometimes detailed documentation written by curators or art historians, but there are limited ways to search for images beyond the given categories. To do so, the user must already know what they are looking for and deploy the correct keywords, both of which require highly specialized knowledge.

However, if we can search the images aligned with their content, then all users will be able to access the database,

and search using a broader and more comprehensive scope. For example, a user could search for all 19th-century French landscape paintings, either winter scenes and summer scenes, with or without figures, etc., and locate all the works in the large database without failing to locate relevant images.

Distributed representation can also be useful to suggest other relevant concepts to user's search words. For example, when we look for a specific book, browsing nearby shelves can sometimes produce a more useful book even if the book is titled with words that we do not initially consider. In art searches, we cannot access the physical storage of the works, but instead we gain information about links between content words, thereby connecting a larger number of art works to our search.

Conclusion

In this paper, we introduced the first deep-learning approach to computationally analyze the contents in fine art paintings. Motivated by significant performances and broad adaptability of deep neural networks in computer vision, we adopted the Inception-V3 as the primary model of our content detector, validated its performance, and considered its last layer parameters as informative resources related to content. In general, the system showed positive correlations with survey responses, but limitations regarding certain types of paintings especially in complex depictions or compositions. To refine our models, we are still looking at other advanced deep-learning algorithms. For example, beyond words, we could build a system to describe art using natural language. A recurrent neural network on top of our system would be a feasible example (Vinyals et al. 2015). Furthermore, the current system perceives the whole image at once, but as content in paintings is often spatially local rather than global, principles in scene labeling (Farabet et al. 2013) or attention modeling (Xu et al. 2015) are expected to provide more sophisticated boards for computational content analysis.

References

Bown, O. 2012. Generative and adaptive creativity: A unified approach to creativity in nature, humans and machines. In *Computers and creativity*. Springer. 361–381.

Carneiro, G.; da Silva, N. P.; Del Bue, A.; and Costeira, J. P. 2012. Artistic image classification: An analysis on the printart database. In *European Conference on Computer Vision*, 143–157. Springer.

Carneiro, G. 2011. Graph-based methods for the automatic annotation and retrieval of art prints. In *Proceedings of the 1st ACM International Conference on Multimedia Retrieval*, 32. ACM.

Crowley, E. J., and Zisserman, A. 2014. In search of art. In *Workshop at the European Conference on Computer Vision*, 54–70. Springer.

Dyke, J. C. V. 1887. *Principles of Art*. New York: New York, Fords, Howard , Hulbert.

Elgammal, A.; Liu, B.; Kim, D.; Elhoseiny, M.; and Mazzone, M. 2018. The shape of art history in the eyes of the

machine. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Elgammal, A.; Kang, Y.; and Den Leeuw, M. 2018. Picasso, matisse, or a fake? automated analysis of drawings at the stroke level for attribution and authentication. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Farabet, C.; Couprie, C.; Najman, L.; and LeCun, Y. 2013. Learning hierarchical features for scene labeling. *IEEE transactions on pattern analysis and machine intelligence* 35(8):1915–1929.

Gombrich, E. H. 1985. Image and word in twentieth-century art. *Word & image* 1(3):213–241.

Hendriks, E., and Hughes, S. 2009. *Van Goghs brushstrokes: Marks of authenticity?*

Jafarpour, S.; Polatkan, G.; Brevdo, E.; Hughes, S.; Brasoveanu, A.; and Daubechies, I. 2009. Stylistic analysis of paintings using wavelets and machine learning. In *Signal Processing Conference, 2009 17th European*, 1220–1224. IEEE.

Kim, D.; Liu, B.; Elgammal, A.; and Mazzone, M. 2018. Finding principal semantics of style in art. *IEEE International conference on semantic computing*.

Lowry, B. 1967. *The visual experience : An introduction to Art*. Englewood Cliffs New Jersey: Prentice-Hall,INC. and HARRY N. ABRAMS, INC.

Mikolov, T.; Chen, K.; Corrado, G.; and Dean, J. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Munsterberg, M. 2009. *Writing about Art*. <http://writingaboutart.org>: New York, Fords, Howard , Hulbert.

Panofsky, E. 1971. *Early Netherlandish painting: its origins and character*, volume 1. Natl Gallery of Art.

Picard, D.; Gosselin, P.-H.; and Gaspard, M.-C. 2015. Challenges in content-based image indexing of cultural heritage collections. *IEEE Signal Processing Magazine* 32(4):95–102.

Polatkan, G.; Jafarpour, S.; Brasoveanu, A.; Hughes, S.; and Daubechies, I. 2009. Detection of forgery in paintings using supervised learning. In *Image Processing (ICIP), 2009 16th IEEE International Conference on*, 2921–2924. IEEE.

Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; and Wojna, Z. 2016. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2818–2826.

Vinyals, O.; Toshev, A.; Bengio, S.; and Erhan, D. 2015. Show and tell: A neural image caption generator. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 3156–3164.

Xu, K.; Ba, J.; Kiros, R.; Cho, K.; Courville, A.; Salakhudinov, R.; Zemel, R.; and Bengio, Y. 2015. Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning*, 2048–2057.