Cognitive Computation (2021) 13:570–582
https://doi.org/10.1007/s12559-021-09823-y

Using Principal Paths to Walk Through Music and Visual Art Style Spaces Induced by Convolutional Neural Networks

E. Gardini[1,2] · M. J. Ferrarotti[1] · A. Cavalli[1,2] · S. Decherchi[1]

Received: 12 August 2020 / Accepted: 5 January 2021 / Published online: 1 February 2021

Abstract
Computational intelligence, particularly deep learning, offers powerful tools for discriminating and generating samples such as images. Deep learning methods have been used in different artistic contexts for neural style transfer, artistic style recognition, and musical genre recognition. Using a constrained manifold analysis protocol, we discuss to what extent spaces induced by deep-learning convolutional neural networks can capture historical/stylistic progressions in music and visual art. We use a path-finding algorithm, called principal path, to move from one point to another. We apply it to the vector space induced by convolutional neural networks. We perform experiments with visual artworks and songs, considering a subset of classes. Within this simplified scenario, we recover a reasonable historical/stylistic progression in several cases. We use the principal path algorithm to conduct an evolutionary analysis of vector spaces induced by convolutional neural networks. We perform several experiments in the visual art and music spaces. The principal path algorithm finds reasonable connections between visual artworks and songs from different styles/genres with respect to the historical evolution when a subset of classes is considered. This approach could be used in many areas to extract evolutionary information from an arbitrary high-dimensional space and deliver interesting cognitive insights.

Keywords Computational intelligence · Convolutional neural networks · Deep learning · Principal path

Introduction

Computational intelligence (CI) and cognitive computations are relatively young fields in science and engineering, with the first ideas dating back to Turing [1]. Some of the most successful CI systems are based on deep artificial neural networks and their variations, which model networks of neurons to solve tasks such as pattern recognition, learning, memorization, and generalization [2]. Recently, the community has become interested in other creative and innovative applications of deep artificial neural networks. These applications include cognitive tasks such as sentiment analysis, neural language processing, neural style transfer, artistic style recognition, and musical genre identification. They require that different network architectures are addressed. Convolutional neural networks (CNNs) [3, 4] are a kind of deep artificial neural network. CNNs are particularly effective for tasks like the recognition, analysis, and classification of images and videos [5–8].

In the field of sentiment analysis, works like [9–11] and many others have tried to transform abstract concepts like emotions into images and sounds. Neural-style transfer applications use CNNs to recombine the content of one image with the style of another, like in [12]. In terms of recognizing styles of visual art, some of the most interesting solutions have been proposed by Lecoutre et al. [13], Karayev et al. [14], and Tan et al. [15], who achieved promising results using two publicly available CNN architectures (AlexNet [7] and ResNet [16]) with the same dataset (Wikipainting).

Recently, two papers [17, 18] addressed the problem of how learning systems *perceive* visual art aspects, such as stylistic properties when compared to human-derived artistic principles.

Regarding the classification of musical genres, Bahuleyan recently proposed a particularly interesting

✉ S. Decherchi
sergio.decherchi@iit.it

[1] Computational & Chemical Biology, Italian Institute of Technology, Via Morego 30, 16163 Genoa, Italy

[2] Dept. of Pharmacy and Biotechnology, University of Bologna, Via Belmeloro 6, 40126 Bologna, Italy

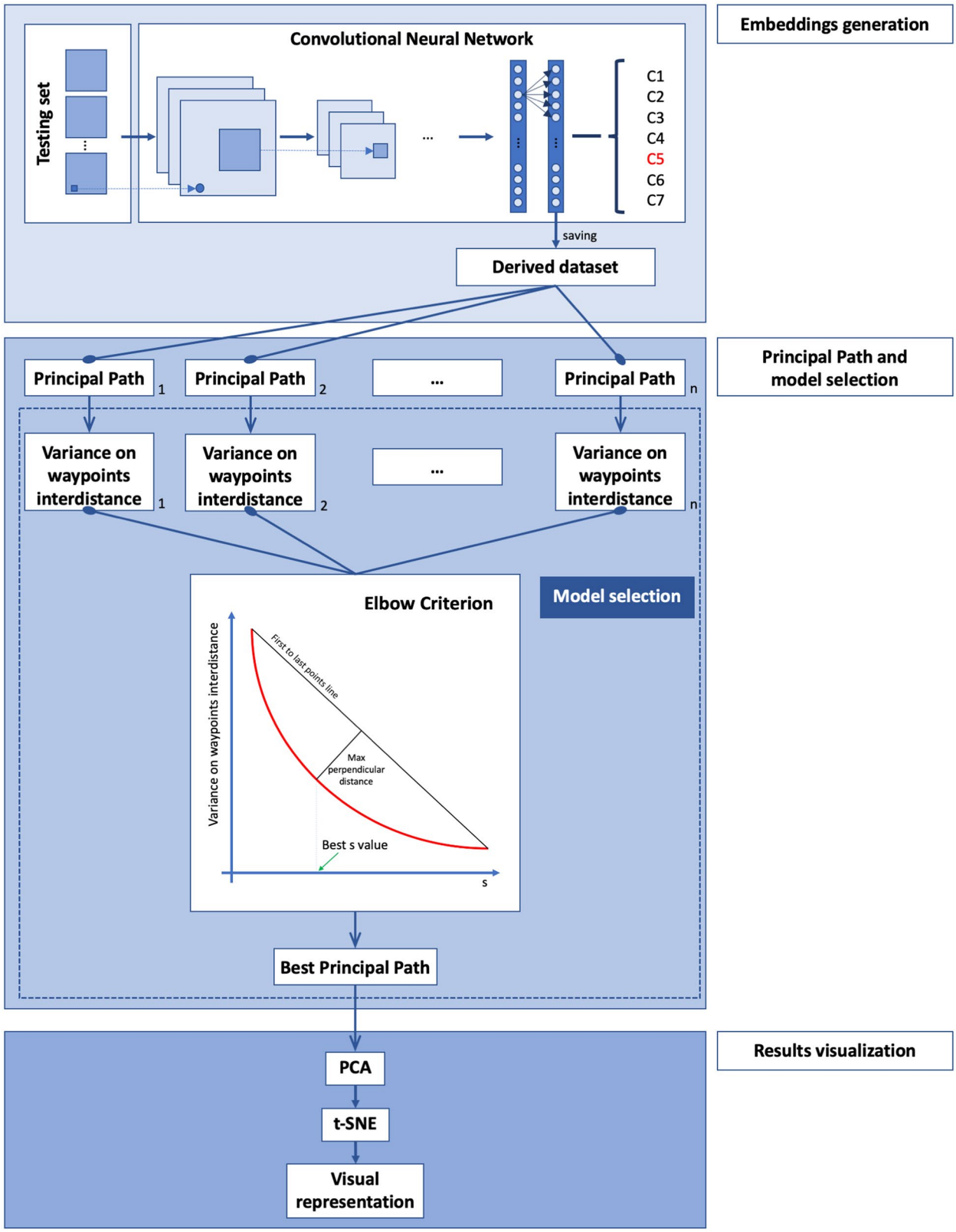

Fig. 1 Schematic representation of the presented protocol. Step 1: featurization with CNNs. Step 2: principal path on the induced vector space with different s values and model selection with the elbow criterion. For each s value, one can obtain a principal path and the corresponding variance on waypoints interdistance for that path. The elbow is the furthest point from the line which connects the first and the last points. Step 3: visualization of the best principal path

solution using CNNs [19]. This approach uses spectrograms (visual representations of the time and frequency information of an audio signal) generated from a subset of songs in Audio Set [20].

Music and visual art share a fundamental trait, namely their historical evolutionary nature. Cultural and historical events influence changes in visual art movements and musical genres. They can thus be considered in evolutionary terms.

Here, we propose to understand and navigate the manifolds induced by previous CNN architectures focusing on the historical evolutionary changes from one song/visual artwork to another. To achieve this, we used a recently devised algorithm [21, 22] to navigate and analyze vector spaces, with a start point and an end point as input. This protocol can be used to assess the quality of the embeddings and to detect relevant transitions and relations within samples [22].

At a technical level, first we extracted features from the penultimate layer of the CNN model to represent each song/visual artwork as a point in an n-dimensional space, then we explored this n-dimensional space. Clearly, the more classes in each dataset, the more complex the problem and the more difficult it is to provide reliable results. This is particularly relevant given that the CNN-induced space is not necessarily optimal for this task. We therefore simplified the problem by considering one subset of classes (styles or genres) at a time. We chose the principal path (PP) [21] as a method to capture the local topology of the data [23].

PP is analogous to a principal curve [24] with predefined start and end points. It can be informally defined as a smooth path connecting two samples and passing through the local support of the data distribution [21]; in other words, we solve a locally constrained manifold learning problem. Being inspired by the minimum free energy path concept of statistical mechanics, PP can be used to infer maximal probability morphing between samples in data space.

The PP is constructed as a discrete curve where intermediate waypoints are connected by straight segments. These waypoints represent subsequent steps in the underlying morphing process; therefore, it is crucial to analyze and visualize them. Here, we classify them by considering their neighbors, and we visually represent them in a low-dimensional embedding obtained through the t-Distributed stochastic neighbor embedding (t-SNE) projection method [25].

In Section 2, we describe each step of our protocol in detail. Section 3 is devoted to the results of our experiments in the music and visual art contexts. In Sections 4 and 5, we discuss possible future application fields of our protocol and offer some final remarks.

Methods

The proposed protocol has three main steps elucidated in Fig. 1:

1. In the first step, a CNN is used as a featurization tool.
2. In the induced vector space, we run the PP algorithm from historically stylistically meaningful end points.
3. Lastly, we visualize the paths and data distribution via t-SNE and comment on the consistency of the recovered waypoints with respect to the true historical evolution in style.

Hereafter, we detail each step of the protocol.

Embedding Generation

The first step of our protocol involves using existing CNN architectures to produce features from their penultimate layer. As mentioned, CNNs have obtained excellent results in large-scale image recognition; they are similar to ordinary neural networks, but with layers of convolving filters applied to local features [26]. Convolutional layers are followed by pooling layers, forming modules. These are followed by fully connected layers, as in the standard feedforward neural networks.

Convolutional layers allow the extraction of features. They comprise neurons arranged in feature maps. Each neuron has a receptive field, which is connected to a neighborhood of neurons in the previous layer by a set of trainable weights. Input images are convolved with the trained weights to create a new feature map. The convolved results are then sent through a nonlinear activation function [6]. Formally, the k-th output feature map Y_k can be computed as:

$$Y_k = f(W_k * x) \tag{1}$$

where x is the input image; W_k is the convolutional filter related to the k-th feature map, $*$ is the 2D convolutional operator, and $f(\cdot)$ represents the nonlinear activation function (sigmoid, hyperbolic, or rectified linear units - ReLUs) [6].

Pooling layers reduce the spatial resolution of the feature maps; max pooling aggregation layers are typically used. They propagate the maximum value to the next layer within the receptive field [6]. Finally, fully connected layers unroll the intermediate maps to get a classical linear feature representation. The softmax operator is usually used for the classification task as the very last neuron for each class [6].

In this work, we used two existing CNN architectures, which are widely used for image classification, namely ResNet-50 [16] and VGG-16 [27].

The VGG-16 architecture differs from the original ConvNet architecture [7] because it has more layers (sixteen), but very small (3x3) convolution filters. As a result, the network has better performance and accuracy. However, increasing the network depth is not always beneficial. Beyond a certain point degradation issues arise and the number of training errors grows larger. Such problem is successfully addressed by the ResNet-50 architecture introducing deep residual learning framework blocks. In particular, if one denotes by $H(x)$ the desired underlying mapping of a few stacked layers, with x being the input to the first of these layers, and if one hypothesizes that multiple nonlinear layers can approximate any function, one can assume that the residual functions $H(x) - x$ can be learnt correctly. The stacked layers can thus approximate a residual function $F(x) = H(x) - x$ and the original function becomes $F(x) + x$. The latter formulation can be realized by feedforward neural networks with shortcut connections, which simply perform identity mapping. Their outputs are then added to the outputs of the stacked layers. This network produces substantially better prediction results than previous networks and is deeper [16].

Table 1 shows the original configurations for VGG-16 [27] and ResNet-50 [16]. Here, we used the original ResNet-50 architecture for the visual art space experiments [28]. For the music space experiments, we used the architecture proposed by Bahuleyan in [19], with the original VGG-16 core connected to a new feedforward neural network. Here, we ignore the predicted labels of the CNNs; instead, we extract the output from the last pooling layer. We used it to represent each entry of the input datasets (songs and visual artworks) as points in an n-dimensional space.

Table 1 VGG-16 and ResNet-50 standard architectures

VGG-16	ResNet50	
16 weight layers	50 weight layers	
Input 224x224 RGB image	Input 224x224 RGB image	
Conv3-64	Conv7-64	
Conv3-64		
Maxpool	Maxpool	
Conv3-128	Conv1-64	X3
Conv3-128	Conv3-64	
	Conv1-256	
Maxpool		
Conv3-256	Conv1-128	X4
Conv3-256	Conv3-128	
Conv3-256	Conv1-512	
Maxpool		
Conv3-512	Conv1-256	X6
Conv3-512	Conv3-256	
Conv3-512	Conv1-1024	
Maxpool		
Conv3-512	Conv1-512	X3
Conv3-512	Conv3-512	
Conv3-512	Conv1-2048	
Maxpool	Avgpool	
FC-4096	FC-1000	
FC-4096		
FC-1000		
Soft-max	Soft-max	

Evolutionary Analysis

The second step of the proposed protocol is to navigate the previously produced vector space using the PP. As previously mentioned, the PP helps extract relevant information about the data topology. More specifically, a smooth morphing between two data samples is obtained solving the following minimization problem:

$$\min_{W,u} \sum_{i=1}^{N} \sum_{j=1}^{N_c} \|\phi(x_i) - w_j\|^2 \delta(u_i, j) + s \sum_{i=0}^{N_c} \|w_{i+1} - w_i\|^2 \quad (2)$$

where N is the number of samples, N_c is the number of waypoints, $\Phi(\cdot) : \mathbb{R}^d \rightarrow \mathbb{R}^{d'}$ is the (possibly nonlinear) transformation mapping of the d-dimensional input space, x_i is a sample/row of the $N \times d$ matrix X, w_j is a waypoint/row of the $N \times d'$ matrix W, and $\delta(u_i, j)$ is the Kronecker delta where u_i are waypoint memberships.

This functional, formally, is an extension of the k-means clustering, where the first and last clusters are constrained (i.e. w_0 and w_{NC+1}) and the other clusters are evolved according to the regular k-means cost function plus the added regularization term, which induces a string topology. All the clusters are waypoints for the path and are topologically connected by a chain of springs [21]. The hyper-parameter s regulates the trade-off between data-fitting and the smoothness of the inferred path. To solve the minimization problem, it is possible to derive an expectation maximization (EM) algorithm, which generalizes the well-known EM for the original k-means. Additionally, being a kernel method, an EM algorithm was also defined without an explicit notion of the transformed space implied by $\Phi(\cdot)$. It is worth stressing that a PP is quite invariant against a large range of N_c values [21]. In our experiments, we arbitrarily set N_c to 50, which we found to be a reasonable value to understand and visualize morphing paths of interest.

In this work, the PP algorithm was run several times with decreasing values of the regularization parameter s on an evenly spaced log scale from 10^5 to 10^{-5}, without changing start and end points. In this way, one obtains several PPs for each pair of start/end samples. To choose the best one, we devised a fast and simple model selection strategy that we explain below. Given a PP, call it W, let $l_{i,i+1} = ||w_i - w_{i+1}||$ be the length of the segment connecting two subsequent waypoints w_i, w_{i+1} and let $Var(l_{i,i+1})$ be the variance of this quantity along the PP.

For large values of s, we expect to get $Var(l_{i,i+1}) = 0$ since the algorithm asymptotically finds the trivial path connecting w_0 to w_{N_c+1} with an evenly sampled straight line. For small value of s, the algorithm instead finds an arbitrary noisy path, usually leading to high values of $Var(l_{i,i+1})$ and collapsing to a regular k-means result.

For each experiment, we decided to select the best PP, W_{best}, using the elbow criterion [29] on the plot of $Var(l_{i,i+1})$ vs s (see model selection in Fig. 1 for further details). In our experience, this simple heuristic is able to select nontrivial paths with equally spaced waypoints, which seamlessly translate into regularly sampled morphing processes.

Path Interpretation and Visualization

The protocol's third step is to analyze the PP obtained in the previous step. Two different strategies to achieve this aim are explored.

The first strategy is to use t-SNE [25] to provide a 2D data visualization of the data points, the waypoints of the PP, and the points of a trivial path. A trivial path is here defined as a mere linear connection of the end points and is used as a reference to prove that results are not obvious. In order to improve t-SNE efficiency, the principal component analysis (PCA) [30] algorithm is used to reduce the dimensionality before t-SNE itself.

Fig. 2 Labeling and 2D visualization of the principal path in visual art space. **a** Labels of the nearest artwork for each waypoint of the principal path (top) and the trivial path (bottom). On the left, results for the following classes: Baroque, Neoclassicism, Realism, Expressionism. On the right, results for the following classes: Early Renaissance, Baroque, Romanticism, Abstract Art. **b** 2D representation of the principal path and the trivial path through four different styles: Baroque, Neoclassicism, Realism, Expressionism (on the left); Early Renaissance, Baroque, Romanticism, Abstract Art (on the right). The x and the y coordinates are the output of the dimensionality reduction performed with t-SNE [25]. The start point and the end point are the most recent and the oldest visual artworks, respectively. The principal path is composed of 50 intermediate points (waypoints) plus the boundaries


a
Principal Path: Baroque, Neoclassicism, Realism, Expressionism
Trivial Path: Baroque, Neoclassicism, Realism, Expressionism
b
Principal Path: Early Renaissance, Baroque, Romanticism, Abstract Art
Trivial Path: Early Renaissance, Baroque, Romanticism, Abstract Art

◂**Fig. 3** Principal path visual artworks. Nearest visual artworks to the waypoints of the principal path and the trivial path, removing consecutive repeated artworks and considering four classes: **(a)** Baroque (blue), Neoclassicism (orange), Realism (green), Expressionism (red); **(b)** Early Renaissance (blue), Baroque (orange), Romanticism (green), Abstract Art (red).

The second strategy is to detect the sample songs and visual artworks nearest to the waypoints of the PP and to the points of the trivial path.

Results

In this section, we report the results of our evolutionary analysis in the visual art and music spaces. The implementation was written in the Python/Keras [31] framework with Tensorflow [32] as backend. The code to reproduce all the experiments is available on github. In particular, users can find the code to perform the featurization at [33, 34] and the code to reproduce the results at [35]. We ran all the experiments on a high performance computing cluster node equipped with a nVidia P100 GPU and 2 Intel Xeon E5-2650 v4 CPUs.

Visual Art Space Experiments

In terms of the visual art space, the manifold induced by the network defined in [13] was analyzed, without performing any retraining procedure. The network implementation is the one in [28] because it was explicitly trained to recognize visual art styles, and because the aim of the present analysis was to understand the relations among them.

As dataset, we used the Wikipainting dataset, which is a large and widely used dataset collection from the WikiArt website [36]. It comprises 80,000 images, each tagged with one of the following 25 styles: *Abstract Art*, *Abstract Expressionism*, *Art Informel*, *Art Nouveau (Modern)*, *Baroque*, *Color Field Painting*, *Cubism*, *Early Renaissance*, *Expressionism*, *High Renaissance*, *Impressionism*, *Magic Realism*, *Mannerism (Late Renaissance)*, *Minimalism*, *Naive Art (Primitivism)*, *Neoclassicism*, *Northern Renaissance*, *Pop Art*, *Post-Impressionism*, *Realism*, *Rococo*, *Romanticism*, *Surrealism*, *Symbolism* and *Ukiyo-e*. The featurization led to 2048 variables from the penultimate layer of the network.

We navigated this space using the PP, selecting three or four classes at a time with a reasonable historical distance to simplify the analysis and using the most recent and the oldest visual artworks as start/end points. We generated a path comprising 50 waypoints plus the start and end points. To understand the transitions, we retrieved the nearest picture for each waypoint. To check the significance of the results and for comparison, we generated a trivial path by connecting the start and end points with a straight line and splitting it into 50 equally distributed points. Figure 2a shows the class (the true style label) of the retrieved nearest pictures for the PP and for the trivial path, considering different styles.

To visualize the points together with the paths, we reduced the dimensions via PCA (sklearn implementation with n_components = 50 and random_state = 5) [30] followed by t-SNE (sklearn implementation with n_components=2, random_state=20, perplexity=50 and learning_rate=300) [25]. PCA reduced the number of features from 2048 to 50, improving the t-SNE algorithm's efficiency (Fig. 2b). Within this simplified and class-reduced setting, the PP recovered the historical evolution of the artistic style together with the content of the visual artworks. This indicates that these CNN-induced spaces at least partially reflect the historical evolution of the styles. In contrast, the trivial path moved blindly and jumped from the start class to the end class without any interesting intermediate. These aspects are emphasized when the start and end points are the historically oldest and newest visual artworks, respectively.

Additionally, we generated other paths by perturbing the start/end points (e.g. using the second/third most recent and the second/third oldest visual artworks as start/end points). The results of these analyses are available in our github repository [35] (folders /results/images/mode=2_1-22-18-25 and /results/images/mode=2_22-14-17-3) and show that the PP is robust to perturbations. The PP's ability to capture the historical evolution of visual art is highlighted when we select and plot the nearest visual artworks to the points of the two paths, as shown in Fig. 3, where repeated consecutive artworks along the path were not shown in this and all the subsequent figures for clarity of representation.

In Fig. 3a, the PP finds visual artworks that depict people and then landscapes. The style varies from black and white to color, with a gradual change from dark cool colors to light warm ones. With the exception of some noisy visual artworks, the evolution of the classes reflects the historical evolution of the art (i.e. Baroque ∼17th–18th century, Neoclassicism ∼18th–19th century, Realism ∼19th century, Expressionism ∼20th century). In contrast, the trivial path passes through two visual artworks that have different styles and content and that belong to the first and last class, respectively. The same gradual morphing can be observed in Fig. 3b. The evolution of the pictures reflects the historical evolution of the art (i.e. Early Renaissance ∼14th–16th century, Baroque ∼17th–18th century, Romanticism ∼ 18th–19th century, Abstract Art ∼20th century), with the colors and content gradually changing from one artwork to the next. There is an interesting jump from romanticism to abstract art, with a strong similarity in terms of shape: the man and the sculpture are both in the center of image, and the top of the sculpture resembles the hat of the man in the portrait. Once again, the trivial path passes through two

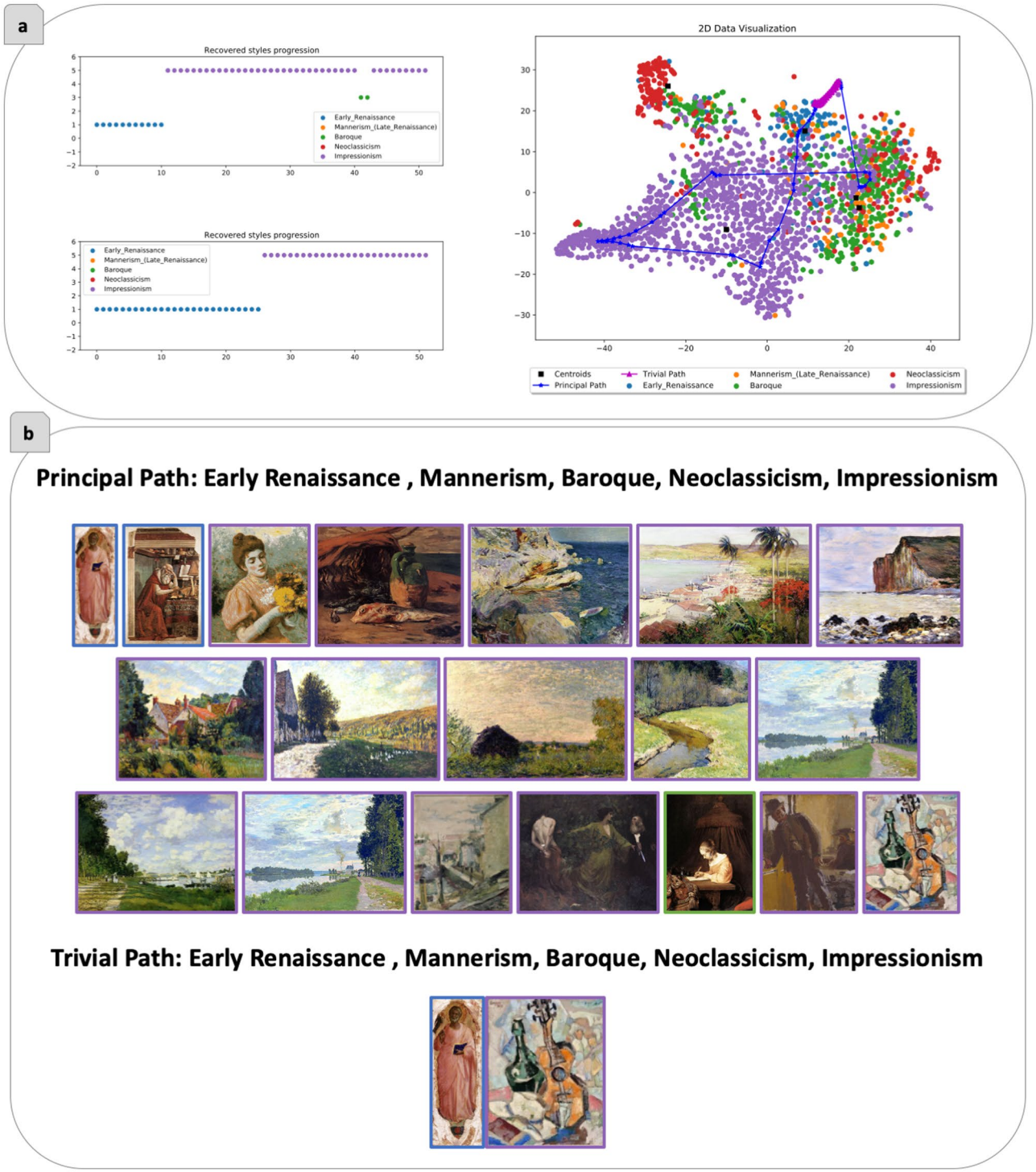

Fig. 4 Historical evolution is not always correctly recovered. **a** On the left, labels of the nearest artworks for each waypoint of the principal path (top) and the trivial path (bottom). On the right, 2D representation of the principal path and the trivial path through four different styles: Early Renaissance, Mannerism, Baroque, Impressionism. The x and the y coordinates are the output of the dimensionality reduction performed with t-SNE [25]. The start point and the end point are the most recent and the oldest visual artworks, respectively. The principal path comprises 50 intermediate points (waypoints) plus the boundaries. **b** Nearest visual artworks to the waypoints of the principal path, removing consecutive repeated artworks and considering four classes: Early Renaissance (blue), Mannerism (orange), Baroque (green), Impressionism (purple)

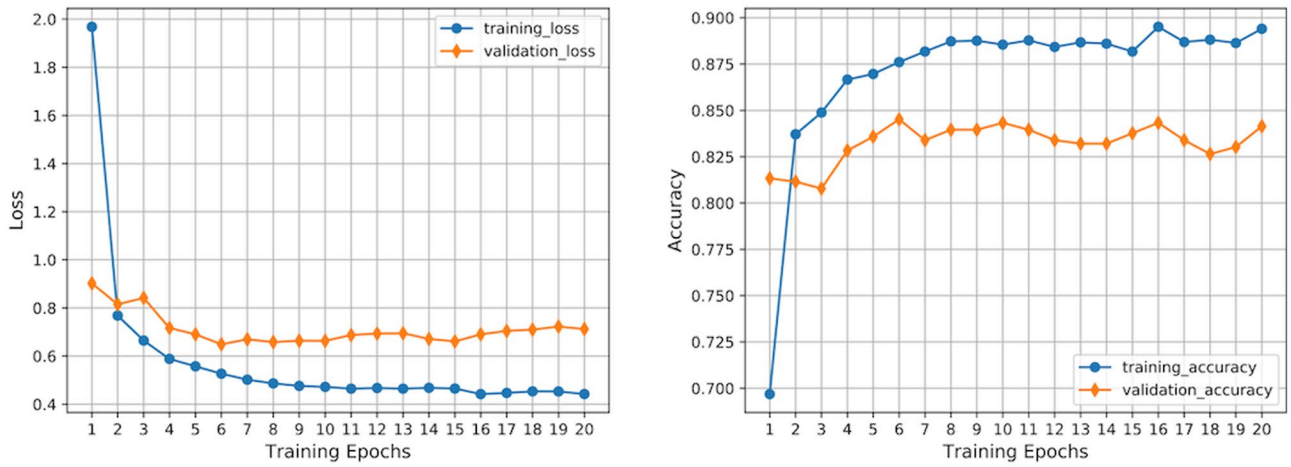

Fig. 5 Learning curves for model selection. Epoch 6 minimizes the validation loss

visual artworks with different styles and content that belong to the first and last class, respectively.

The historical evolution is not always found by the PP solution, as shown in Fig. 4. In this case, among a subset of five classes (Early Renaissance, Mannerism, Baroque, Neoclassicism and Impressionism), the PP retrieved only visual artworks belonging to the Early Renaissance and the Impressionism, with a little deviation to the Baroque. Even if the historical evolution is not respected, the PP is clearly able to perform a gradual morphing from the start to the end points. This is particularly clear when compared to the performance of the trivial path, which once again passes through two visual artworks with different styles and content that belong to the first and last class, respectively. Further results in the visual art space are freely available in our github repository [35]. For example, we provide the results of experiments with different subsets of classes, where the start/end points can be selected manually by visual inspection, set as the most recent and the oldest visual artworks, or set as the centroids of the clusters that correspond to the youngest and oldest historical periods in the subset of classes/styles.

Table 2 Principal path vs. trivial path: music space experiments. Number of songs found by the principal path and the trivial path along their way (without duplicated samples and grouped by classes). Baroque, Opera, Classical, Jazz, Rock (on the top), Medieval, Baroque, Jazz, Rock (on the bottom)

	Principal path	Trivial path
Baroque	3	1
Opera	3	1
Classical	8	3
Jazz	2	-
Rock	11	1
Total	27	6
	Principal path	Trivial path
Medieval	4	1
Baroque	6	0
Jazz	6	1
Rock	16	5
Total	28	7

Music Space Experiments

In the music space, we repeated the above experiments, analyzing the manifold induced by the network defined in [19] and available at [37]. This time, we trained the network to recognize music genres. In contrast to the visual art context, the literature contains music experiments that used datasets with very different traits. The best-known datasets with audio tracks include RWC (465 entries) [38, 39], GZTAN genre (1000 entries) [40], Magnatagatune (25863 entries) [41], and AudioSet (40540) [20]. We used the Magnatagatune dataset [41] for our music experiment instead of the AudioSet [20] used by [19]. We have chosen this dataset as it is one of the largest with available audio tracks; additionally it is enriched with tag annotation files that assign a list of genres and instruments to each song. Specifically, we selected all the songs belonging to one of the following genres: Baroque, Classical, Jazz, Medieval, Opera, Rock. We established a univocal class for each song by choosing the most specific genre (e.g. a song labeled as Baroque and Classical becomes Baroque). In this case, the featurization led to 512 features from the penultimate layer of the network.

We randomly split the dataset into training (80%) and test (20%) sets. The validation set was 10% of the training set. We trained the model for no more than 20 epochs using the same batch size (100) and using the ADAM optimizer (default learning rate = 0.001)[42]. Figure 5 shows the

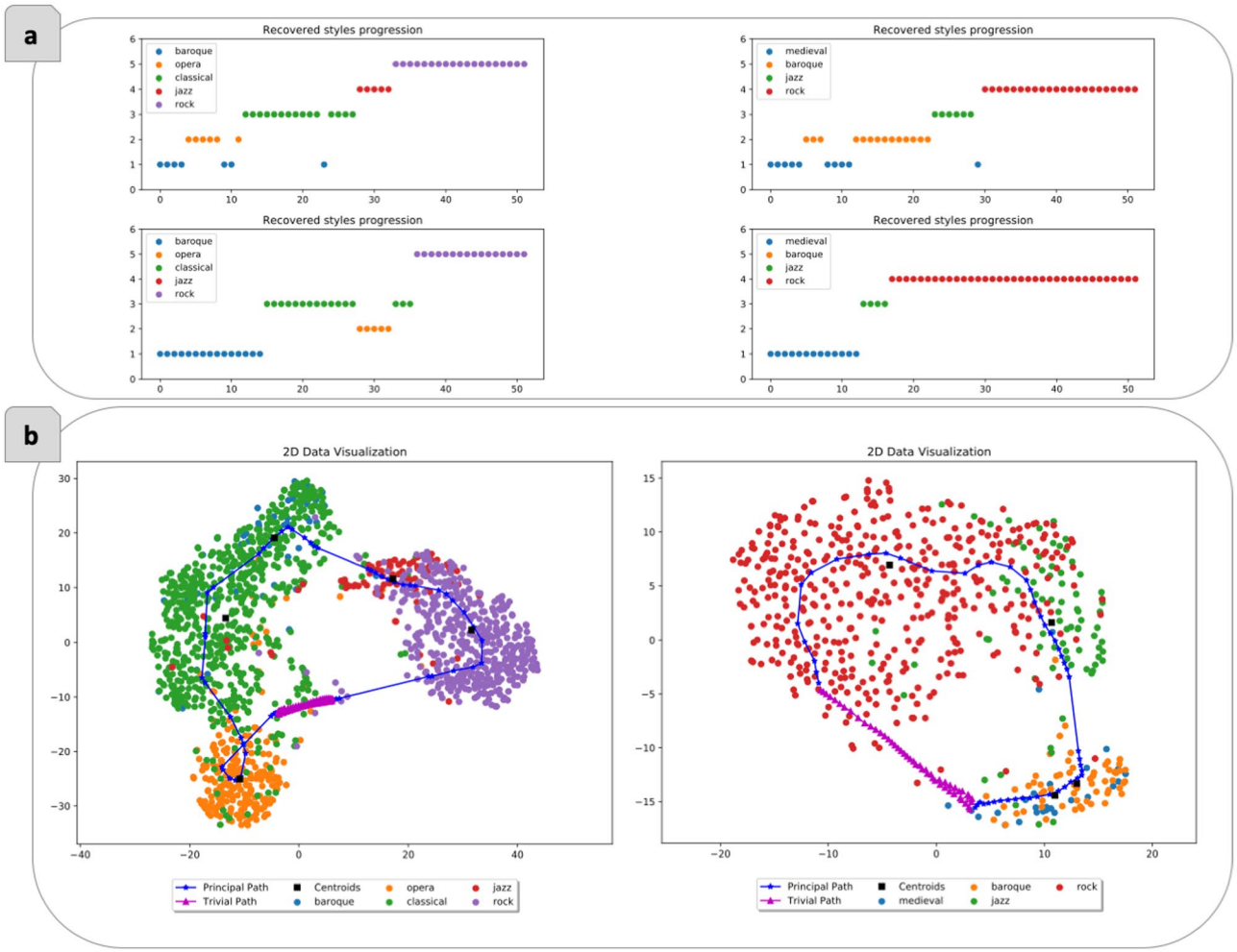

Fig. 6 Labeling and 2D visualization of the principal path in music space. **a** Labels of the nearest songs for each waypoint of the principal path (top) and the trivial path (bottom). On the left, results for the following classes: Baroque, Classical, Opera, Jazz, Rock. On the right, results for the following classes: Medieval, Baroque, Jazz, Rock. **b** 2D representation of the principal path and the trivial path through three different styles: Baroque, Classical, Opera, Jazz, Rock (on the left); Medieval, Baroque, Jazz, Rock (on the right). The x and the y coordinates are the output of the dimensionality reduction performed with t-SNE [25]. The start point and the end points are selected visually. The principal path comprises 50 intermediate points (waypoints) plus the boundaries

learning curves of the model. We selected the best model as the one obtained at the sixth epoch. This model had the highest accuracy and the lowest loss on the validation set. Finally, we tested the best model on the test set for accuracy, F-score, area under the curve (AUC) and Matthews correlation coefficient (MCC), obtaining 0.86, 0.55, 0.9, and 0.77, respectively.

Again, we navigated these spaces using the PP and the trivial path, considering three or four classes at a time with a reasonable historical distance, with the start/end points selected visually and generating 50 intermediate waypoints. Figure 6a compares the class variation for the nearest song to each waypoint (as class we considered the true style label of each nearest song). Figure 6b shows the 2D visual representation of the two paths. This was obtained by reducing the number of features from 512 to 50 with the PCA algorithm (sklearn implementation with n_components=50 and random_state=5) [30], then reducing the number of features from 50 to 2 with the t-SNE algorithm (sklearn implementation with n_components=2, random_state=20, perplexity=50 and learning_rate=300) [25]. We have omitted the resulting spectrograms because they are difficult to interpret. The corresponding songs are listed in our github repository [35], and Table 2 shows a summary of the ones we found.

Figure 6a highlights the PP's ability to navigate the space, taking into account the evolution of the musical genres,

despite the complexity of the problem. The trivial path seems to find intermediate genres between the start and end classes. However, analyzing Table 2, one can observe that the PP performs a smooth transition finding different songs along its way. In contrast, the trivial path finds few songs and the transition from the start point to the end point is not gradual.

Other examples using different classes and start/end points are freely available in our github repository [35].

Discussion

The principal path concept was inspired by the minimum free energy path in statistical mechanics as a principle to define a path in space. Here, we have shown that, in a simplified reduced-class setting, this tool coupled with CNNs can capture the evolutionary/historical connections in the image/music space. This simplified setting has been used because the topological value of the CNN-induced spaces is suboptimal for this task. Further research can be done in this direction. The findings, however, might suggest that many evolutionary processes could be studied by approaching them as a minimum free-energy path-finding problem. This would not be the first time that machine learning and statistical mechanics had profound points of connections, with the Boltzmann Machine [43] and Boltzmann generators [44] being prominent examples. In terms of cognition, this raises the question of whether minimum free energy paths are how humans connect ideas. If we generalize the concept of start and end points to ideal objects (ideas), then one could ask whether the way we think can be formalized as an attempt to move from one idea to another via maximal probability moves (namely minimum free energy regions, or regions which bear many ideas, as probability is linked to the number of points/ideas in the space) [45]. This ultimately is connected to the notion of cognition and intelligence, if one defines intelligence as *reading through things* (from Latin *intus legere*), that is, connecting ideas via paths. The conjecture is that maximal probability paths, the ones we seek with the PP algorithm, can describe many phenomena including idea morphing, which is ultimately a creative cognitive process. It would be interesting to understand to what extent this principle is general and what it can retrieve when applied to image/song spaces, as done here.

The presented model is implicitly generative due to the presence of waypoints, even though a probability density function is not available. The explicit creation of new objects would require reversing the CNN representation; this could be achieved in future via variational autoencoders for instance. We used a nearest neighbor strategy to associate an existing song or visual artwork to each waypoint and to analyze the morphing from the start point to the end point.

As noted above, this first attempt has some limitations, mainly because the shape of the CNN manifolds is less than ideal for this task, thus preventing the method's full application to several classes. Music and visual art are just two of the many potential fields of application. In the field of sentiment analysis and affective computing, the PP concept has been used to inspect the topology of the concept distribution in the embedding space, as shown in [22, 46].

In the music context, our method could work as a playlist creator. In fact, the PP is able to provide a list of songs that are a gradual morph from one initial song to another. The PP can thus work as a recommendation system for music. A similar idea based on a different algorithm is presented in [47].

Our method could also be useful in any context where it makes sense to reason in terms of evolutionary connections between data points, or in terms of a pseudo-time that connects data points in succession. One interesting example would be quantitative biological datasets (e.g. transcriptomics or metabolomics) to better understand time-dependent phenomena, such as tumor evolution, cell cycle, cell differentiation, and organogenesis, which is an area on which we are working. In particular, the method could potentially be used to identify transition states between different tumor stages, and thus to identify molecular markers of cancer progression. This would have important clinical implications for the early detection of tumors or staging in general. Additionally, given the recent ascent of single-cell resolution in transcript quantification (scRNA-Seq), our method could potentially be used to identify transitional states in processes such as induced differentiation, transition between cell cycle phases, and the onset of drug resistance mechanisms. Efforts in some of these areas are already underway [48].

Conclusion

In this paper, we combined CNN-induced vector spaces and the principal path (PP) concept to navigate the spaces of music and visual art. Along the paths, we identified waypoints that represent a gradual morphing from the start point to the end point. Based on our results, the PP found reasonable connections between visual artworks and between songs from very different genres, partially respecting their historical evolution in a simplified setting. In future, we believe that this approach could be used to perform experiments in many different application contexts in which an evolutionary analysis makes sense.

Acknowledgements We thank Grace Fox for proofreading.

Funding Open Access funding provided by Istituto Italiano di Tecnologia

Compliance with Ethical Standards

Conflicts of Interest The authors declare that they have no conflict of interest.

Ethical Approval This article does not contain any studies with human participants or animals performed by any of the authors.

References

1. Turing AM. Computing machinery and intelligence. In Robert Epstein, Gary Roberts, and Grace Beber, editors, Parsing the Turing Test. Springer Dordrecht. 2009:23–65
2. Engelbrecht AP. Computational intelligence: an introduction. John Wiley & Sons; 2007.
3. LeCun Y, Boser B, Denker JS, Henderson D, Howard RE, Hubbard W, Jackel LD. Backpropagation applied to handwritten zip code recognition. Neural Comput. 1989;1(4):541–51.
4. LeCun Y, Boser BE, Denker JS, Henderson D, Howard RE, Hubbard WE, Jackel LD. Handwritten digit recognition with a back-propagation network. In D. S. Touretzky, editor, Advances in Neural Information Processing Systems. Morgan-Kaufmann. 1990;2:396–404.
5. LeCun Y, Bengio Y, Hinton G. Deep learning. Nature. 2015;521(7553):436–44.
6. Rawat W, Wang Z. Deep convolutional neural networks for image classification: A comprehensive review. Neural Comput. 2017;29(9):2352–449.
7. Krizhevsky A, Sutskever I, Hinton GE. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, Advances in Neural Information Processing Systems. Curran Associates, Inc. 2012;25:1097–1105.
8. Russakovsky O, Deng J, Su H, Krause J, Satheesh S, Ma S, et al. Imagenet large scale visual recognition challenge. Int J Comput Vis. 2015;115(3):211–52.
9. Mood of the planet. https://sentic.net/mood-of-the-planet.pdf. Accessed 02 Dec 2020.
10. Sensity. https://stanza.co.uk/sensity/. Accessed 02 Dec 2020.
11. Illuminations. http://vibeke.info/illuminations. Accessed 18 Sept 2020.
12. Jing Y, Yang Y, Feng Z, Ye J, Yu Y, Song M. Neural style transfer: A review IEEE. Trans Vis Comput Graph. 2020;26(11):3365–85.
13. Lecoutre A, Negrevergne B, Yger F. Recognizing art style automatically in painting with deep learning. In Min-Ling Zhang and Yung-Kyun Noh, editors, Proceedings of the Ninth Asian Conference on Machine Learning of Proceedings of Machine Learning Research. PMLR. 2017;77:327–342.
14. Karayev S, Trentacoste M, Han H, Agarwala A, Darrell T, Hertzmann A, Winnemoeller H. Recognizing image style. 2014. http://arxiv.org/abs/1311.3715
15. Tan WR, Chan CS, Aguirre HE, Tanaka K. Ceci n'est pas une pipe: A deep convolutional network for fine-art paintings classification. In 2016 IEEE International Conference on Image Processing (ICIP) 2016;3703–3707.
16. He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2016;770–778.
17. Cetinic E, Lipic T, Grgic S. Learning the principles of art history with convolutional neural networks. Pattern Recogn Lett. 2020;129:56–62.
18. Elgammal A, Liu B, Kim D, Elhoseiny M, Mazzone M. The shape of art history in the eyes of the machine. In Proceedings of the 32nd AAAI conference on Artificial Intelligence. 2018;2183–2191.
19. Bahuleyan H. Music genre classification using machine learning techniques. 2018. http://arxiv.org/abs/1804.01149.
20. Gemmeke JF, Ellis DPW, Freedman D, Jansen A, Lawrence W, Moore RC, Plakal M, Ritter M. Audio set: An ontology and human-labeled dataset for audio events. In 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). 2017;776–780.
21. Ferrarotti MJ, Rocchia W, Decherchi S. Finding principal paths in data space. IEEE Transactions on Neural Networks and Learning Systems. 2019;30(8):2449–62.
22. Ragusa E, Gastaldo P, Zunino R, Ferrarotti MJ, Rocchia W, Decherchi S. Cognitive insights into sentic spaces using principal paths. Cogn Comput. 2019;11(5):656–75.
23. Carlsson G. Topology and data. Bull Am Math Soc. 2009;46(2):255–308.
24. Hastie T, Stuetzle W. Principal curves. Journal of the American Statistical Association. 1989;84(406):502–16.
25. van der Maaten L, Hinton G. Visualizing data using t-SNE. Journal of Machine Learning Research. 2008;9(86):2579–605.
26. Lecun Y, Bottou L, Bengio Y, Haffner P. Gradient-based learning applied to document recognition. Proc IEEE. 1998;86(11):2278–324.
27. Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. 2014. http://arxiv.org/abs/1409.1556.
28. bnegreve/rasta. https://github.com/bnegreve/rasta. Accessed 04 Dec 2020.
29. Thorndike RL. Who belongs in the family? Psychometrika. 1953;18(4):267–76.
30. Jolliffe I. Principal component analysis. In Miodrag Lovric, editor, International Encyclopedia of Statistical Science. Springer Berlin Heidelberg, Berlin, Heidelberg. 2011;1094–1096.
31. Fran Sois Chollet et al. Keras. 2015. https://keras.io.
32. Abadi M, Agarwal A, Barham P, Brevdo E, Chen Z, Citro C, et al. TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. 2015. https://www.tensorflow.org/.
33. Image featurization. https://github.com/erikagardini/ImageFeaturization. Accessed 04 Dec 2020.
34. Music featurization. https://github.com/erikagardini/MusicFeaturization. Accessed 04 Dec 2020.
35. Using principal path to walk through music and visual art style spaces induced by convolutional neural networks. https://github.com/erikagardini/Using-PP-to-walk-through-music-and-visual-art-style-spaces-induced-by-CNN. Accessed 04 Dec 2020.

36. Wikiart.org - visual art encyclopedia. https://www.wikiart.org/. Accessed 04 Dec 2020.
37. Recognizing the genre of music files using machine learning and deep learning models. https://github.com/HareeshBahuleyan/music-genre-classification. Accessed 04 Dec 2020.
38. Goto M, Hashiguchi H, Nishimura T, Oka R. RWC Music Database: Popular, classical and jazz music databases. In Proceedings of the 3rd International Conference on Music Information Retrieval (Ismir). 2002;2:287–288.
39. Goto M. Development of the rwc music database. In Proceedings of the 18th International Congress on Acoustics (ICA). 2004;1:553–556.
40. Tzanetakis G, Cook PR. Musical genre classification of audio signals. IEEE Transactions on Speech and Audio Processing. 2002;10(5):293–302.
41. The magnatagatune dataset |city university mirg. http://mirg.city.ac.uk/codeapps/the-magnatagatune-dataset. Accessed 04 Dec 2020.
42. Kingma DP, Ba J. Adam: A method for stochastic optimization. 2014. http://arxiv.org/abs/1412.6980.
43. Salakhutdinov R, Hinton G. Deep boltzmann machines. In David van Dyk and Max Welling, editors, Proceedings of the 20th International Conference on Artificial Intelligence and Statistics. 2009;5:448–455.
44. Noé F, Olsson S, Köhler J, Wu H. Boltzmann generators: Sampling equilibrium states of many-body systems with deep learning. Science. 2019;365(6457):eaaw1147.
45. Friston K. The free-energy principle: a unified brain theory? Nat Rev Neurosci. 2010;11(2):127–38.
46. Cambria E, Li Y, Xing F, Poria S, Kwok K. Senticnet 6: Ensemble application of symbolic and subsymbolic ai for sentiment analysis. In Proceedings of the 29th ACM International Conference on Information & Knowledge Management. Association for Computing Machinery 2020;105–114.
47. Boil the frog. http://boilthefrog.playlistmachinery.com. Accessed 04 Dec 2020.
48. Gardini E, Giorgi FM, Decherchi S, Cavalli A. Spathial: an R package for the evolutionary analysis of biological data. Bioinformatics. 2020;36(17):4664–7.