Analyzing Art Works: The Six Steps Methodology

Luis Fernando Gutiérrez and **Rafael Pérez y Pérez**
Departamento de Ingeniería Industrial, Universidad de los Andes, Bogotá, Colombia
lf.gutierrez399@uniandes.edu.co
Departamento de Tecnologías de la Información, UAM Cuajimalpa, México City.
rperez@correo.cua.uam.mx

Abstract

Art analysis is a key aspect for computational systems whose goal is to generate visual artifacts. This paper proposes a six steps methodology to analyze and represent design principles from art works. Our approach starts with an image segmentation followed by the construction of a straight skeleton. Then we extract some color information and perform shape analysis and classification. Finally some design principles are calculated and groups of elements are built over a complete graph. Our internal art work representation gives us a way to approximate the phases of artistic appreciation proposed by some authors. We show a procedure to generalize compositional rules for the generation of new abstract art works based on the steps of the proposed methodology. We plan to use a self organizing map to cluster our art work representations and use this information to build a hypergraph and/or multigraph. Since these graphs can represent design principles, the system will be able to use these structures to explore new ways to generate artifacts and measure their novelty compared to previous exemplars.

Introduction

As suggested by Cetinic et al. (2018), analyzing artworks is a complex task which generally involves understanding aspects like form, content and meaning. These aspects originate from the formal elements present in the artwork such as line, shape, color, texture, mass and composition (Barnet, 2015). Art experts usually do their analysis comparing paintings to find relations between them (Seguin et al., 2016). Generally, the outcomes of those analyses can lead to style classifications, genre determinations, formal comments and influences between artists, artworks or art movements (Saleh et al. 2014, Florea et al. 2017, Badea et al., 2018).

In the last few years, research in computer vision techniques to analyze visual art has increased in quantity and quality (Badea et al., 2018). This trend depends on two facts. First, there have been consistent efforts by museums and collectionists to digitize more paintings and include relevant meta-data. This permits us to have larger datasets to do analysis. The second fact is the development of deep neural networks. Style classification has received more attention (Bar et al., 2015; Saleh and Elgammal, 2015; Elgammal et al. 2018). Genre classification has been explored but there are still complex open challenges (Condorovici et al., 2013). The approach used in such classificators, usually includes extracting a set of image features and using them to train different classifiers such as support vector machines, neural networks or k-nearest neighbors (Cetinic et al., 2018).

Art analysis is a key aspect for computational systems whose goal is to generate visual artifacts. Such analysis allows building knowledge structures from real pieces of art that can be exploited by creative agents to create more elaborated outputs. Thus, we need to develop mechanisms that allow computer systems to improve their "art appreciation" in general (Norton et al., 2010; Health et al., 2016). In order to accomplish this goal, we require general and specific knowledge (Barret, 2007). That is what this project is about. We are interested in studying and representing notions related to composition and design principles like balance, symmetry, size, contrast and shape (Pérez y Pérez et al. 2013; Pérez y Pérez & Guerrero 2019) from abstract pieces of visual art. *In this paper, we claim that the development of systems that allow analyzing and representing design principles from well-known pieces of art are important to generate better creative agents*. Thus, we propose a six steps methodology (6SM) that combines and advances well known algorithms for image processing in order to obtain such design principles.

This is a work in progress. Therefore, the key target of this text is to present to the reader the core aspects of our six steps methodology to represent design principles (see the section titled Art Work Representation) as well as showing some partial results. The first step consists of an image segmentation using the algorithm proposed by Syu et al. (2017) (see the Image Segmentation section). The objective of this phase is to build a hierarchical multi-resolution representation of the regions that make up an image. The second step builds a planar graph called straight skeleton over every region or segment of the previous step (see the Straight Skeleton and Centrality Measure section). The purpose of this graph is to induce a terrain model. With this model a centrality measure can be computed and a generalized notion of center of mass can be defined. The third step extracts color information (see the section titled Color Information) based on Itten's model (1974). To achieve this phase, we follow Sartori et al. (2015) proposal to build a 180 color swatch. With this color palette we replace every original color and calculate spatial relations over Itten's color sphere generated by these 180 colors. The fourth step consists of a shape classification process (see the Shape Classification section). The objective of this step is to do a clustering procedure to have a reduced number of general shapes that can represent most of the regions of the art works analyzed. The next phase implements some binary relations between regions based on some design principles (see the Design Principles and Binary Relations section). We calculate relations on pairs of shapes based on measurements over the mean shape bounding box, direction and aspect ratio. The final step consists on building groups as suggested by (Pérez y Pérez et al. 2013; Pérez y Pérez and Guerrero 2019) (see the section titled Groups of Regions). The goal of this step is to have a representation of the elements of the art work at different levels of abstraction.

We are employing the Tlahcuilo visual composer (Pérez y Pérez et al., 2013; Pérez y Pérez & Guerrero, 2019) to implement a proof of concept. We will include new design principles and new evaluation procedures. Because having more artistic knowledge should help achieve higher quality artifacts (Heath et al. 2016), we hope to improve the quality and novelty of the artifacts the Tlahcuilo produces.

Related Work

Recently, a large number of image representations presented in the literature are exclusively or highly dependent on abstract neural network feature maps. Of particular interest to this research are works that suggest image representations such as the one proposed by Bar et al. (2014) that involve more concepts interpretable by humans directly. The authors suggest using a combination of powerful neural network visual features with other descriptors. The effectiveness of convolutional neural network based features, particularly in combination with other hand-crafted features, was confirmed also for genre classification by Cetinic and Grgic (2016).

Artistic style classification is another related problem that has been addressed with continuous increasing interest. Some recent work used object recognition (Crowley and Zisserman, 2014). The authors show that finding objects in paintings by learning object-category classifiers from available sources of natural images is possible. Artistic scene or genre understanding is also important. Badea et al. (2018) investigate the relation between genre, scene and artistic image subject. The authors investigate abstraction achieved by deep convolutional neural networks. In particular, "Abstract Art", is targeted by the authors as a challenging problem since a subject is not necessarily present.

Saleh et al. (2014) study how painters influence each other using visual similarity. They implemented a procedure based on computer vision and machine learning. The authors perform several comparisons using different visual features and similarity measurements. Since there is not enough ground truth information to achieve influence analysis directly, the authors use a highly correlated task such as style classification to show some results. The authors investigate features aspect of the paintings and compare semantic-level features vs low-level and intermediate level features. They claim that their study confirms that high-level semantic features are more useful for style classification and hence for influence analysis. In a similar direction, Seguin et al. (2016) investigate how state-of-the-art machine vision algorithms can be used to retrieve common visual patterns shared by sets of paintings. Florea et al. (2017) suggest that visual similarity has space for improvement because most of the research and results have been developed for older artistic movements where scene depiction has high-level semantic concepts and does not present particular abstractions.

In the domain of computational creativity, DARCI (Norton et al. 2010; Heath et al. 2016) is a reference. The goal of this system is to eventually produce images through creative means. In the process to achieve this, the authors propose to teach DARCI some artistic image appreciation and understanding. They implement this through the association of low-level image features to artistic descriptions. They show that the system successfully learns 150 different descriptors from images. Pérez y Pérez and his colleagues (Pérez y Pérez et al. 2013; Pérez y Pérez & Guerrero 2019) propose a computer model to develop visual compositions based on the Engagement and Reflection Model. The system uses design principles to analyze examples provided by designers and generate a knowledge base to progress a visual work and also measure the novelty of its artifacts.

Garcia and Vogiatzis (2018) argue that to build artistic knowledge, we have to work outside the style classification tasks and expand our research goals. The authors present SemArt, a multi-modal dataset for semantic art understanding. The authors suggest a challenge called Text2Art to evaluate art understanding based on a retrieval task. They also suggest several models for encoding visual and textual artistic representations into a common semantic space. Strezoski and Worring (2018) created a large dataset with more than 430,000 samples called "The OnmiArt Challenge". They suggest analyzing more attributes related to the art works.

Art Work Representation

The knowledge structures constructed in this paper are based on examples of abstract art created by human artists. Since abstract art can be thought of as lacking representation of common everyday objects, these art works are more prone to intrinsic artistic formal aspects. Initially we propose to develop an analysis using exclusively 165 of Rothko's art works. We have a partial prototype that implements the pipeline described in the next few sections. It receives an image as input and outputs our internal representation.

Step 1: Image Segmentation

Syu et al. (2017), claim that even though plentiful segmentation algorithms have already been proposed, how to effectively partition an image into segments that are “meaningful” to human visual perception is still very challenging. Sometimes it is not enough to choose a specific image characteristic such as color or texture to achieve a successful image segmentation. This paper proposes to use the new algorithm developed by Syu et al. (2017) which builds a hierarchical image segmentation. One of the objectives of the algorithm is to generate a dendrogram in which every node corresponds to a segment and all the nodes in the same level build up a segmentation. This dendrogram is a consistent multi-resolution representation of the contents of the segmented image. In our context, consistent means that every new segment comes exclusively from a previous node. This characteristic is one of the main differences between the algorithm developed by the authors and similar hierarchical solutions.

The algorithm proposed by Syu et al. (2017) works in two steps like almost all hierarchical procedures. The first phase works directly with the raw pixels and has the objective to group up similar pixels into regions. For the second step of the first phase of the algorithm, the authors present an iterative process of contraction and merging. The contraction step is based on an optimization process over an affinity matrix that groups pixels. The merging is characterized by a fixed grid in which previously contracted pixels are joined together. These two steps work exclusively on the similarity at color level between adjacent pixels.

The second phase keeps on making a contraction and merging procedure. The affinity matrix gets updated based on color, size, texture and intertwining of the regions. The affinity matrix for the second phase depends on the dissimilarity between regions R_i and R_j. The factors that make up the dissimilarity metric are the following:

Color Component: For a region R_i, Syu et al. (2017) denote its color feature as the averaged color values inside the region. For adjacent regions R_i and R_j, the color-based dissimilarity measure is $D_C(R_i, R_j) = \| C_{R_i} - C_{R_j} \|_2$.

Texture Component: two adjacent regions with similar texture patterns and similar colors should have a larger affinity value. To describe the texture pattern of a region, Syu et al. (2017) convert the region color values to gray and calculate the Weber Local Descriptor (WLD). This descriptor consists of two components, differential excitation and orientation, over a local window around every pixel.

Region Size Component: To take into account the region size in the merging process of the second phase, the authors define an additional distance function is design to facilitate the merging of two small regions or the merging of small regions into their neighboring regions fast.

Spatial Intertwining Component: this component is used to merge together small regions produce during the cycles. Syu et al. (2017) measure the intertwining of any pair of regions i and j based on a fixed 5x5 window over every pixel p of R_i. They calculate the most common index in the local window. Based on these indexes, a decision to merge smaller groups of pixels to neighboring regions is taken. Figure 1 and 2 give examples of the result of the first phase of the process we are describing. We show the hierarchical segmentation procedure achieved by our partial prototype applied to one of Rothko’s and Kandinsky’s artworks.

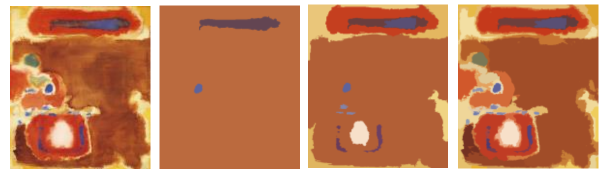

Figure 1 – original, 3, 13 and 60 segments

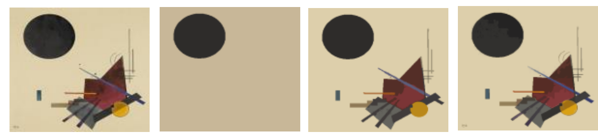

Figure 2 – original, 2, 13 and 60 segments

Step 2: Straight Skeleton and Centrality Measure

According to Huber (2012), the notion of a straight skeleton for a simple polygon P was introduced for the first time by Aichholzer et al. (1995). The authors used a wavefront propagation process to define it. As reported by Huber (2012), every edge e of P sends out a wavefront which moves inwards at unit speed and parallel to e. Figure 3 shows a visualization of the described process. During the wavefront propagation process, topological changes named events are produced. Huber distinguishes two types: edge and split events. An edge event occurs when two neighboring convex vertices u and v of the wavefront meet. This event causes the wavefront edge e, which connects u and v, to collapse to zero length. The wavefront edge e is removed and the vertices u and v are merged into a new convex vertex. A split event occurs when a reflex vertex u of the wavefront meets and edge e of the wavefront. The vertex u splits the entire wavefront into polygonal parts (Huber, 2012).

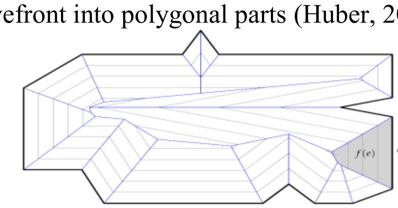

Figure 3 – visualization of the wavefront propagation process. The edge e has the associated front wave $f(e)$. Image reproduced from Huber (2012), pg. 13.

The formal definition of a straight skeleton taken from Huber (2012) is as follows: the straight skeleton $S(P)$ of a polygon P corresponds to the straight segments that are traced out by the vertexes of any wavefront. These segments are denominated arcs or edges of $S(P)$. The places where topological changes take place are defined as nodes. To every edge e of P belongs a face $f(e)$, which consists of every point traced by the wavefront border started by edge

e. Every node of $S(P)$ is incident to multiple arcs. The border of any given face consists of the arcs and vertices of $S(P)$.

The straight skeleton has several interesting properties. Central to this work is the fact that this skeleton is a tree (Aichholzer et al., 1995). Based on that fact, we know there exists a unique node that can be named as the root. The same authors generalized the concept of straight skeletons to planar straight-line graphs. According to Huber (2012), this generalization allowed a better interpretation of the visualization suggested initially by Aichholzer et al (1995). This intuition is formalized with the definition of model terrain: the terrain $T(G)$ of G a straight-line graph is: $T(G) = \bigcup_{t\geq 0} W(G,t) * \{t\}$ were $W(G,t)$ corresponds to the wavefront of a straight-line graph G for a time $t \geq 0$. According to Huber (2012), $W(G,0)$ corresponds geometrically to the same graph G and it should be visualized like a superposition with the same topology over the original graph.

The straight skeleton induces an Euclidean graph given by the spatial coordinates of the nodes. Taking into account the terrain model induced by the straight skeleton and the distance between every node, a weight for every edge is assigned. The value is just the multiplication of the $t \geq 0$ parameter or third dimensional value of $T(G)$ with the distance between every pair of nodes. We calculate a centrality measure based on closeness and using the weights as cost function to find a unique node that can represent each region. Since the straight skeleton is a tree, we can be certain that such a unique node always exists. The center of this Euclidean graph, can represent each region and acts as a generalized notion of a mass center. Figure 4 shows the terrain model induced by the straight skeleton of a region and visualizes the center of the Euclidean graph.

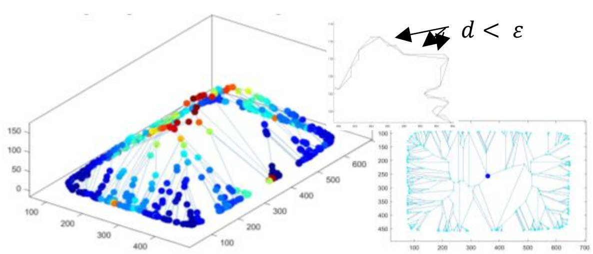

Figure 4. Terrain model induced by a straight skeleton. Center of Euclidean graph based on closeness centrality measure.

The process by which the straight skeleton is generated is computationally demanding. To cope with this time and the large number of regions used in this paper, we propose to make a region simplification based on the Ramer–Douglas–Peucker algorithm. The objective of the algorithm is, given a linear segment curve, to find an approximated curve with less points. The dissimilarity measure used to compare both curves is based on the Hausdorff distance. One of the most important characteristics of the procedure is that the approximated curve points, come from a subset of the original curve points. For the algorithm to work, a tolerance parameter $\varepsilon > 0$ needs to be given by the user. After some trial and error, we found $\varepsilon = 0.1$ to be a good candidate. The first step of the algorithm is to find the farthest points on the original curve. Once both points of the maximum diameter are identified, the region is split in two curves and the algorithm is invoked recursively on both segments. The original two points are automatically assigned as elements of the new approximated curve. The next step is to find the farthest point between the straight line made up by the two original points and every other point of the segment. Once this point is identified, the distance between the line and the point is calculated. If the distance is less than ε then any given points between the original points and the farthest point get eliminated. This guarantees that the tolerance requirement is met in every step. In case the distance is greater than ε, the algorithm marks this new point as belonging to the new approximated curve. The algorithm gets recursively invoked between the new subsegments. The recursion process is over when there are no more segments to check. Setting $\varepsilon = 0.1$ allows us to simplify every border region without losing important perceptual geometrical characteristics. After the simplification based on the Ramer–Douglas–Peucker algorithm, we order all the components of every segmentation in every level. In case we find more than one component, we only calculate the straight skeleton over the largest component. If holes are found inside any of the regions, we calculate their size and include them only if they represent more than 1% of the original containing region.

Step 3: Color Information

Most of the color information is going to be based on Itten's theory (1974). Itten studied and taught almost all aspects related to aesthetic and expressivity of color during several years in the Bauhaus school. His theory defined a set of rules for colors and combinations of them. These principles were named by the author as "objective principles of color". Additionally, Itten also tried to formalize some of the emotional aspects of color combinations. The author's theory is visually represented by a color wheel. This structure is composed of 12 color shades made out of primary, secondary and tertiary colors. Colors that are opposed in the color wheel are complementary and make up a harmonic pair (Sartori et al., 2015). Itten's wheel was further expanded using 5 levels of luminance and 3 levels of saturation. This last model, composed of 180 colors was named by the author as Itten's color sphere (Sartori et al., 2015).

In order to condense and reduce color information, we implement a color swatch construction process, based on what Sartori et al. (2015) suggested. The idea of this process is to sample all the colors out of a dataset and use a cluster process like k-means to find 180 centroids in RGB color space. In our case, the dataset is based on all the color regions of every hierarchical segmentation level. The similarity metric used by the k-means algorithm in our case is the ordinary three dimensional Euclidean distance. Once the 180 centroids are found (Figure 5), we proceed to replace every color with the nearest centroid to change every color in terms of the newly built color swatch (Figure 6).

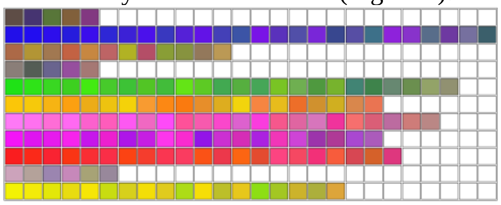

Figure 5. Color palette – 180 colors. 11 groups of colors. Up to down: black, blue, brown, grey, green, orange, pink, purple, red, white and yellow.

Figure 6. Color palette application.

The color relationships used in this paper are going to be binary and related to contrast and harmony. For the first relation, we simply use Itten's color sphere and try to locate opposite elements. For harmony relations, we suggest using the Itten's relations that correspond to defined geometric patterns over the sphere or color wheel. The color swatch also permits us to establish some groups of colors that give global information of the subset of colors from the swatch that are present in an artwork.

Step 4: Shape Classification

The definition of the concept of "shape" has always been complicated. According to Dryden & Mardia (2016), in the ordinary and common use of the word "shape", we almost always use it in an indirect way and using relations of similarity to other objects to try to specify a specific "shape". Dryden & Mardia (2016) present the following definition: "shape is all the geometrical information that remains when location, scale and rotational effects are removed from an object" (pg. 22).

Based on this definition, the shape of an object is invariant under any Euclidean similarity transformation. Two objects have the same shape if one of them can be translated, rescaled and rotated in such a way that superimposing it over the other one, they match completely. The way in which a shape is represented is fundamental to developing any analysis. This work is going to use a finite set of points over the straight skeleton to build some pseudo-landmarks. The configuration matrix X is the kxm matrix of Cartesian coordinates of the k landmarks in m dimensions. In particular, we are going to work with $k \geq 3$ landmarks and the dimension m will be 2. Kendall (1984) demonstrated that for the particular case in which m is 2, the shape space is a Riemannian manifold (complex projective space). In particular and for the purposes of this paper, we need to use the Riemannian distance to be able to compare the difference between any two shapes once any translation, scale or rotation is removed.

To classify the regions generated by the hierarchical segmentation algorithm, we suggest a process that builds several configuration matrixes. The construction starts with $k = 12$ and ends as soon as one of the following two conditions are met: a) the maximum number of points of the region is less than 50, b) ${}^{alr}/_{alo} \geq 0.90$ and $1 \geq {}^{ar}/_{ao} \geq 0.90$, where alr is the arc-length of the approximated curve, alo is the arc-length of the original curve, ar is the area of the approximated curve and ao is the area of the original curve.

We compare every pair of regions based on a 12-points configuration matrix and gradually increase the number of configurations if necessary. We use the Ramer–Douglas–Peucker algorithm to simplify every region on our dataset to find the respective configuration matrixes. In applying the algorithm, we use the notion of tortuosity: $\tau = \frac{L}{C}$ where L corresponds to the length of the curve and C corresponds to the distance between the extreme points. Since every region is closed, we start by splitting every border by the major axis and using the tortuosity measure to decide what sub curve has more complexity and start the procedure. On every step, we compare the tortuosity of every subsegment to decide the next candidate. Figure 7 shows some steps of the above procedure for one region. In case a region has less than 12 points, we artificially interpolate points between the longest segments until necessary.

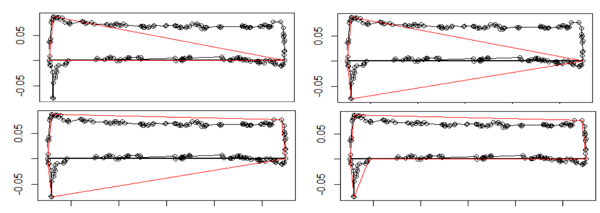

Figure 7. Region 2 of Rothko's "Violet, Black, Orange, Yellow on White and Red" (1949) work. 3, 4, 5 and 6 points simplification in red.

To classify these simplified regions, we do a clustering procedure over all the 12 points configuration matrixes using the ideas of Vinué et al. (2014). The authors suggest an extension to the original k-means algorithm so that it can be applied directly over configuration matrixes using the Procrustes analysis and the Riemannian distance. Starting with 1553 simple regions, we suggest to find initially 90 centroids on this first step. In the second step of our procedure, we do a hierarchical clustering with single linkage to have the possibility of reducing further the number of clusters. To try to identify where the hierarchical tree should be cut, we analyze the distance matrix calculated with the Riemannian distance function between all the centroids. After some experimentation using between 2 and 10 nearest neighbors of every element in this matrix, we found that taking 25% of the maximum distance between any element was a good condition to stop the pruning procedure. We check the condition “b” stated before (${}^{alr}/_{alo} \geq 0.90$ and $1 \geq {}^{ar}/_{ao} \geq 0.90$) to decide if every centroid is a good representative of the cluster. In case there is a configuration region which does not fulfill this condition, we propose to split the respective cluster augmenting the configuration matrixes one step at a time until condition “b” is true. After doing this procedure, the hierarchical clustering tree allows us to move up or down in the representation of any region. If we want to simplify the configuration matrix to generate simpler regions, we use the tree and all the members of a cluster to pick a simpler configuration matrix or generate a new mean shape. The final centroids act as the color swatch we defined before and is going to allow us to generalize some binary relations mentioned in the next section. Figure 8 shows a contrast relation between two regions and gives us an idea of how a generalization of this relation is possible.

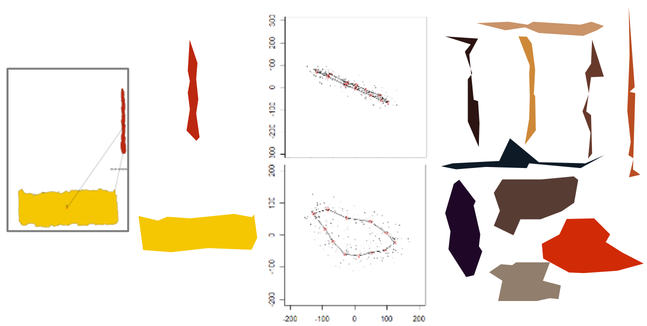

Figure 8. Contrast binary relation – directional, size and proportion. 12-points simplification, cluster centroid and similar regions for relation generalization.

Step 5: Design Principles and Binary Relations

Based on the output of the previous step, we are going to define some contrast binary relations between the centroids. After a linear transformation that takes the end points of the major axis of every centroid to the coordinates $\left(-\frac{1}{2}, 0\right)$ and $\left(\frac{1}{2}, 0\right)$ we calculate the mean shape bounding box, aspect ratios, distance and mean direction. To define this direction, we propose to use a histogram based on the shortest path that connects the major axis end points and passes through the center of the straight skeleton (Figure 9). We suggest having 6 bins in the histogram. Every bin is 30º and the range starts in -15º. Once the histogram is normalized, we can have an approximation to the orientation of the longest path and use this as a general notion of direction for the region.

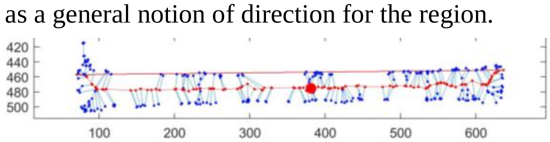

Figure 9. Major axis and shortest path on SS.

This work also proposes to calculate balance, symmetry and rhythm relations as suggested by Pérez y Pérez et al. (2013). We also propose to expand the use of contrast from color (Pérez y Pérez et al. 2010) to shape and incorporate more design principles than the original work developed by the authors. We believe the region representation suggested will allow us to generate even more art principles in future works since the straight skeleton lets us calculate internal symmetries of the shape, proportions and more. In Figure 10 we show some examples of binary relations found using our partial prototype.

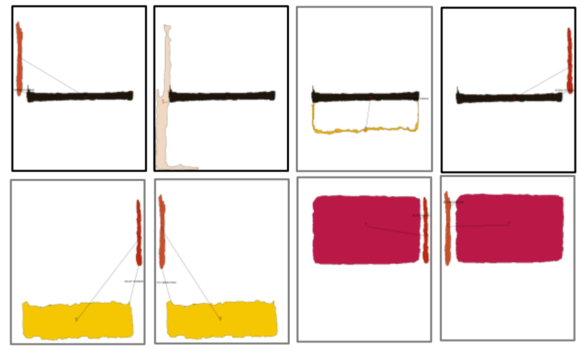

Figure 10. Binary relations – directional, size and proportion contrast examples. Balance and Symmetry.

Step 6: Groups of Regions

Based on the ideas presented in (Pérez y Pérez et al. 2013; Pérez y Pérez and Guerrero 2019), we build groups of regions based on the distance. We construct a complete graph of the first 20 regions of an art work using the normalized center of every region (Figure 11). The order in which the graph is built is based on the size. We calculate the distance between every pair of nodes. The final weight of every edge is the normalized Euclidean distance by the main diagonal of the image of the art work. To start building the groups of layer 1, we follow the procedure described by Pérez y Pérez et al. (2013) and iterate their procedure until no more groups or layers are possible.

Figure 11. Rothko’s “Violet, Black, Orange, Yellow on White and Red” (1949). 10 simple regions and complete graph (SS).

Internal Representation

The goal of the internal representation is to extract the information of the art work related to the design principles and binary relations suggested for the analysis. We suggest a categorical-numerical vector that represents the art work with the following structure:

a) Global information: artist name, year of creation and the artistic style of the art work.
b) Specific art work information: width, height, aspect ratio, first 20 simple regions based on the dendrogram of

segmentation ordered by area in decreasing order. In case there are less regions, values are filled with zeros.

c) Region specific information: every region is represented by the following attributes: area of the region normalized by the total area of the art work, centroid identifier of the cluster to which this region is closer based on Riemannian distance, normalized width of the bounding box of the region, histogram of direction and color identifier from the color swatch.

d) Groups of Regions: information based on the groups constructed using the procedure described previously.

e) Relations between Regions: using the complete graph induced by the center of every one of the simple regions, we propose to analyze every possible binary relation between every pair of regions. Existence of the respective binary relation between any pair of simple regions is represented by 1 or 0. The order of the attributes is based on literal b. Every binary relation is weighted by the distance between the nodes of the complete graph.

Discussion and Future Work

In this paper we have introduced what we refer to as the Six-Steps Methodology (6SM) to analyze and represent design principles from well-known pieces of abstract visual art. The knowledge representation was built taking into account some important aspects of the appreciation and perception processes such as color, orientation, shape, proportion, contrast size and grouping (Liu et al., 2017). In the near future, we plan to include texture more explicitly and work some design principles based on it. The hierarchical segmentation gives us a way to approximate the phases of artistic appreciation proposed by Tinio (2013) and Leder (2013). In particular, the first simple regions can capture the initialization phase mentioned by Tinio, while the rest of the levels of the segmentation can be associated to the expansion, adaptation and finalization phases proposed by the author. And since our representation allows us to get a global structure, recognize and simplify shapes, build groups and get further details as needed, we think that some of Leder's ideas (2013) related to knowledge, familiarity, content and style processing stages are captured too. We believe these facts are important for the development of computational creative systems that generate visual artifacts, because they allow them to analyze information better and constantly move between general and specific knowledge, that is necessary throughout the evaluation of the creative process. We hope to be moving in the right direction to improve the artistic appreciation of our system so it can be even more autonomous.

We have shown a procedure to classify shapes that allow us to generalize compositional rules for the generation of new abstract art works that impact directly the novelty of the artifacts generated by the system. As far as we know there are no other systems capable of obtaining, from human made abstract visual art, design principles that then can be used for generating new pieces. Based on the work reported in this paper, we expect to be able to incorporate all these algorithms into the agent Tlahcuilo visual-composer in the following months. Then, we will be able to test the quality of the new products generated by the system and hopefully produce more results to support our view.

Besides describing the core components of the 6SM, we also suggest that the work reported here can be useful as a base to produce more robust systems. For instance, we plan to use a self organizing map to cluster our art work representations, probably using a variation of the Growing Hierarchical Self Organizing Map (GHSOM) (Rauber et al., 2002). With the cluster's information and temporal data from the art works, we suggest to build a hypergraph and/or a multigraph. Because these graphs can represent design principles, the system will be able to exploit such information, find correlations and explore new ways to generated new pieces (Hackett 2016).

The information represented by these graphs might be also useful to improve the automatic evaluation of visual pieces. For instance, it is possible to compare the hypergraph of a creative agent's products against some recognized visual art. Based on the GHSOM we can check how many previous examples share similar compositional principles and guide the creative agent's composition process depending on these results. In a similar way, depending on the distance to other nodes in the hypergraph and/or the multigraph, we could evaluate novelty. Finally, we believe these structures can be used by the system to give some explanations as to why a partial or finished composition is interesting, what design principles it is based on, what previous stablished rules it might break and probably to what style or styles it belongs.

We are also comparing the results of some state-of-the-art style classification neural networks feature maps with our art work representation to see if it is possible to improve the performance of those models or ours. We require more experiments to produce more hypotheses about how computational systems can generate interesting abstract artifacts that are grounded in an artistic context. The methodology we describe here is one step towards answering deeper questions about how a system uses previously available knowledge in the quest for producing new creative visual artifacts.

References

Aichholzer, O., Alberts, D., Aurenhammer, F. and Gärtner, B. 1995. Straight Skeletons of Simple Polygons. In Proceedings of the 4th International Symposium of LESMARS, 114-124. 1995

Badea, M., Florea, C., Florea, L., & Vertan, C. 2018. Can we teach computers to understand art? Domain adaptation for enhancing deep networks capacity to de-abstract art. Image and Vision Computing, 77, 21-32.

Bar, Y., Levy, N., & Wolf, L. 2014. Classification of Artistic Styles Using Binarized Features Derived from a Deep Neural Network. Computer Vision - ECCV 2014 Workshops Lecture Notes in Computer Science, 71-84.

Barnet, S. 2015. A short guide to writing about art. Pearson.

Barrett, T. 2007. Teaching Toward Appreciation in the Visual Arts. International Handbook of Research in Arts Education Springer International Handbook of Research in Arts Education, 639-656.

Cetinic, E., Lipic, T., & Grgic, S. 2018. Fine-tuning Convolutional Neural Networks for fine art classification. Expert Systems with Applications, 114, 107-118.

Condorovici, R. G., Florea, C., & Vertan, C. 2013. Painting Scene Recognition Using Homogenous Shapes. Advanced Concepts for Intelligent Vision Systems Lecture Notes in Computer Science, 262-273.

Crowley, E. J., & Zisserman, A. 2014. In search of art. In ECCV workshops: 1 (pp. 54–70). Springer.

Dryden, I. L., and Mardia, K. V. 2016. Statistical shape analysis with applications in R. Chichester: John Wiley & Sons.

Elgammal, A., Bingchen, L., Mohammad, E., & Marian, M. 2017. CAN: Creative Adversarial Networks Generating "Art" by Learning About Styles and Deviating from Style Norms. Paper presented at the 8th International Conference on Computational Creativity (ICCC), Atlanta, GA, USA, June 19–23.

Elgammal, A., Mazzone, M., Liu, B., Kim, D., & Elhoseiny, M. 2018. The Shape of Art History in the Eyes of the Machine. arXiv preprint arXiv:1801.07729 (2018).

Florea, C., Badea, M., Florea, L., and Vertan, C. 2017. Domain Transfer for Delving into Deep Networks Capacity to De-Abstract Art. Image Analysis Lecture Notes in Computer Science, 337-349.

Garcia, N. and Vogiatzis, G. 2018. How to Read Paintings: Semantic Art Understanding with Multi-Modal Retrieval. arXiv preprint arXiv:1810.09617 (2018).

Hackett, P. M. 2016. Psychology and philosophy of abstract art: Neuro-aesthetics, perception and comprehension. London: Palgrave Macmillan.

Heath, D. and Ventura, D. 2016. Before a computer can draw, it must first learn to see. In Proceedings of the 7th International Conference on Computational Creativity, 2016.

Huber, S. 2012. Computing straight skeletons and motorcycle graphs: Theory and practice. Aachen: Shaker.

Itten, J. 1974. The art of color: The subjective experience and objective rationale of color. Wiley.

Kendall, D. G. 1984. Shape manifolds, Procrustean metrics and complex projective spaces. Bulletin of the London Mathematical Society, 16, 81–121.

Leder, H. (2013). Next steps in neuroaesthetics: Which processes and processing stages to study? Psychology of Aesthetics, Creativity, and the Arts, 7(1), 27-37.

Liu, J., Lughofer, E., & Zeng, X. (2017). Toward Model Building for Visual Aesthetic Perception. Computational Intelligence and Neuroscience, 2017, 1-13

Norton, D., Heath, D. and Ventura, D. 2010. Establishing Appreciation in a Creative System. In Proceedings of the 1st International Conference on Computational Creativity, 2010.

Pérez y Pérez, R., Aguilar, A., & Negrete, S. (2010). The ERI-Designer: A Computer Model for the Arrangement of Furniture. *Minds and Machines,* 20(4), 533-564.

Pérez y Pérez, R., María González de Cossío, M. and Iván Guerrero, I. 2013. A Computer Model for the Generation of Visual Compositions. *Paper presented at the 4th International Conference on Computational Creativity (ICCC),* Sydney, Australia.

Pérez y Pérez, R. and Guerrero I. R. (2019). A computer agent that develops visual compositions based on the ER-model. *Annals of Mathematics and Artificial Intelligence*.

Rauber, A., Merkl, D., and Dittenbach, M. 2002. The growing hierarchical self-organizing map: Exploratory analysis of high-dimensional data. IEEE Transactions on Neural Networks, 13(6), 1331-1341.

Saleh, B., Abe, K., Arora, R. S., & Elgammal, A. 2014. Toward automated discovery of artistic influence. Multimedia Tools and Applications, 75(7), 3565-3591.

Saleh, B., Elgammal, A. 2015. Large-scale classification of fine art paintings: Learning the right metric on the right feature. arXiv preprintarXiv:1505.00855 (2015).

Sartori, A., Culibrk, D., Yan, Y., and Sebe, N. 2015. Who's Afraid of Itten. Proceedings of the 23rd ACM International Conference on Multimedia - MM 15, 311-320.

Seguin, B., Striolo, C., Dilenardo, I., & Kaplan, F. 2016. Visual Link Retrieval in a Database of Paintings. Lecture Notes in Computer Science Computer Vision – ECCV 2016 Workshops, 753-767.

Strezoski, G., and Worring, M. 2018. OmniArt. ACM Transactions on Multimedia Computing, Communications, and Applications, 14(4), 1-21.

Syu, J., Wang, S., and Wang, L. 2017. Hierarchical Image Segmentation Based on Iterative Contraction and Merging. *IEEE Transactions on Image Processing*, 26(5).

Tinio, P. P. L. (2013). From artistic creation to aesthetic reception: The mirror model of art. *Psychology of Aesthetics, Creativity, and the Arts,* 7(3), 265-275.

Vinué, G., Simó, A., and Alemany, S. 2014. The k-means algorithm for 3D shapes with an application to apparel design. *Advances in Data Analysis and Classification*, 10(1), 103-132.