Bauhaus-Universität Weimar
Faculty of Media
Degree Programme Computer Science And Media

Fourier transform: 3D visualization and Painting classification

Master's Thesis

Jiaqi WENG — Matriculation Number: 115131
Born Nov. 15, 1985 in Shanghai, China

Supervisor: Prof. Dr. Charles Wüthrich

1. First Referee: Prof. Dr. Charles Wüthrich
2. Second Referee: PD Dr. Günther Schatter

Submission date: September 13, 2017

Declaration

Unless otherwise indicated in the text or references, this thesis is entirely the product of my own scholarly work.

Weimar, September 13, 2017

Jiaqi WENG

Abstract

This thesis presents a 3D visualization tool of the Fourier transform(FT) and a method of painting classification using FT. 3D visualization is popular when the computer graphics develops. It offers a totally different view for user to analyze the Fourier transform and perform filtering in the frequency domain.

Fourier transform is an effective tool for pattern recognition. Meanwhile, the painting classification using image analysis is a challenging problem. It is possible to analyze paintings by the Fourier transform and obtain a better result with other methods.

In this work, we attempt to extract features of paintings from the frequency domain. During the course of the thesis, we investigate the brushstroke information and extract features by performing Fourier analysis on the brushstroke texture images of a painting, and we apply machine learning to a dataset sample to perform the classification experiment. Finally, we analyze the results of the experiments and discuss some possible future work to evaluate the method.

Contents

List of Figures

Chapter 1

Introduction

In this thesis, the majority of my study focuses on Fourier transform for digital image analysis. First topic is to create a three-dimensional(3D) visualization tool of the Fourier transform for image processing, illustrating the use of some filters(e.g., low pass, high pass and Gaussian) on the images in 3D. And then, we will use the Fourier transform to solve a real world problem – "painting classification" which is to distinguish paintings from different artist.

1.1 Overview

The Fourier transform is an important image processing tool which is used to decompose an image into its sine and cosine components. The output of the transformation represents the image in the Fourier or frequency domain, while the input image is the spatial domain equivalent. In the Fourier domain image, each point represents a particular frequency contained in the spatial domain image.

The Fourier transform is used in a wide range of applications, such as image analysis, image filtering, image reconstruction and image compression. Three-dimensional visualization tools could help users to see more details in the Fourier transform, and improve the user interaction with the image transform, however, until now there are are only some simple tools in this field(e.g. Matlab). In this project, we will create a tool of 3D visualization of Fourier transform, and then, illustrate the use of some filters for the images in 3D space.

In addition, the problem of artist identification seems ripe for the use of image processing tools. The brushwork is shown as a visual assessment of the presence of the artist's "handwriting". This suggests that analysis of the brushwork could assist the art expert in the process of authentication. Meanwhile, Fourier transform is a powerful tool for texture pattern recognition. It is possible to analyze the brushstroke texture image by Fourier transform, and in this project, we will create a framework based on the Fourier transform of the brushstroke texture to classify the paintings to one of several artists.

1.2 Goals, contributions and outline

Here, the goal of our project is to make a 3D visualization tool of the Fourier transform for image processing to enhance the user interaction when performing the filters of image in the frequency domain. Then, we will use Fourier transform tools for "painting classification", which distinguish paintings by the same painter from one another.

We aims at improving image processing in the frequency domain and making it understandable, then showing a visualization tool in 3D environment. Finally, we will explore the potential of the Fourier transform for analyzing art painting textures.

The main contributions of the project are:

1. The 3D visualization of Fourier transform and some filters.

2. Extracting specific brushstrokes automatically is a challenging problem. This project introduce a brushstroke extraction algorithm based on the Fourier transform.

3. A numerical feature is proposed to characterize brushstrokes.

4. An automatic framework for assessing the level of distinction between paintings.

The remainder of the thesis is organized as follows: First, chapter 2 will present the principles of Fourier transform; some common filters(e.g., low pass, high pass and Gaussian) and the 3D visualization of the Fourier transform. Then, chapter 3 will discuss the problem – "painting classification" including several related methods, the details of the approach and experiments of painting classification of a dataset including six different painters. Finally, chapter 4 provides conclusions to the thesis.

Chapter 2

3D visualization of image transform

In this Chapter, we will discuss the principle of Fourier transform, 3D visualization of Fourier transform and filtering of image in the frequency domain.

2.1 Fourier transform of Image

The Fourier transform of image (in this case, the 2D Fourier transform, abbreviated as FT) is that one image can be expressed as the sum of many basis sine and cosine functions (figure 2.3)(ImageMagick, 2014).

The definitions of the transform and the inverse transform for an image(N*N) are given below:

$$F(k,l) = \sum_{a=0}^{N-1} \sum_{b=0}^{N-1} f(a,b) e^{-\iota 2\pi (\frac{ka}{N} + \frac{lb}{N})} \tag{2.1}$$

and

$$f(a,b) = \frac{1}{N^2} \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} F(k,l) e^{\iota 2\pi (\frac{ka}{N} + \frac{lb}{N})} \tag{2.2}$$

In equations, $f(a,b)$ shows the value of the image in the spatial domain, $F(k,l)$ is the value in the frequency domain and the exponential term ($e^{\pm\iota 2\pi (\frac{ka}{N} + \frac{lb}{N})}$) represents the basis function of the point (k,l) in the Fourier space. The $\frac{1}{N^2}$ is a normalization term in the inverse transform. This normalization is sometimes applied to the forward transform instead of the inverse transform, but it should not be used for both.

Next, Fourier transform(FT) tries to represent all images as a weighted sum of two-dimensional orthogonal basis-like(cosine-like) images. Therefore images that only contain the pattern of pure cosines have particularly simple FT images.

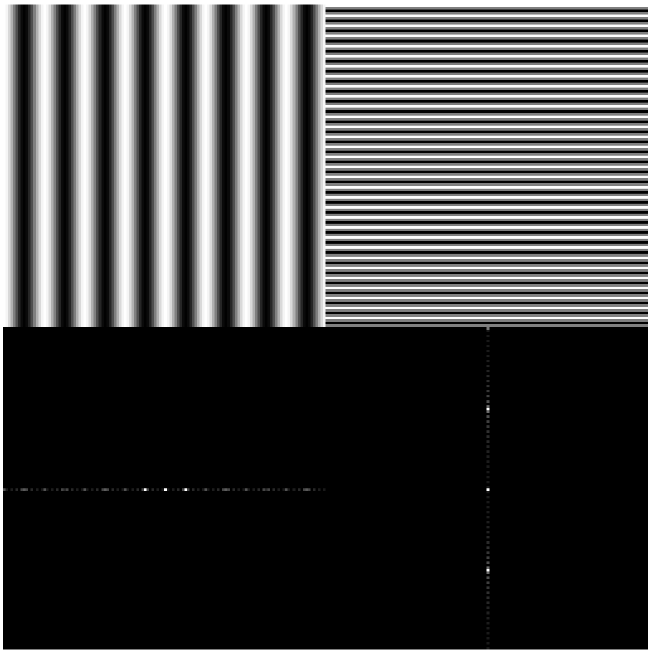


FIGURE 2.1: Fourier transform of cosine-like image

Figure 2.1 shows two images with their Fourier transforms directly underneath. The images are pure horizontal cosine of 8 cycles and a pure vertical cosine of 32 cycles. The result of the FT for each has a single component, represented by 2 bright spots symmetrically placed about the center of the transform image. The center of the FT image is the origin of the frequency coordinate system. The x-axis from left to right represents the horizontal component, and the y-axis from bottom to top represents the vertical component. And the center of FT image represents the average value of the image.

Images usually have a large average value and many low frequency information so FT images usually have bright blob of components near the center. Positions of the Fourier transform image closer to the center, represent the lower frequencies. Comparing with the two FT examples, the right one has higher frequencies than left one, and the result shows that the distance of the bright dots away from the center is further.

As the definition above, $f(a, b)$ is the image and is real, but $F(k, l)$ (abbreviate as F) is the Fourier transform and is complex. Generally, F is represented by its real and imaginary part (or magnitude and phase). Briefly, the magnitude tells the quantity of the frequency components and the phase tells the position of the frequency components.

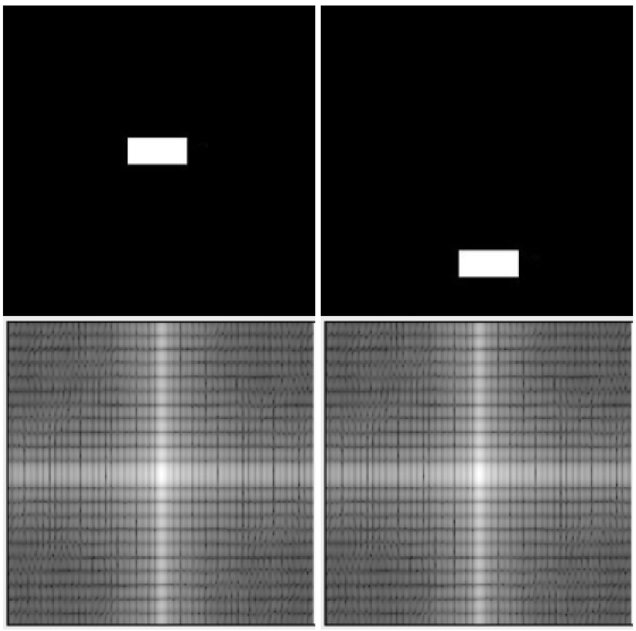


FIGURE 2.2: Magnitude image of Fourier transform shows the quantity of the frequency component

Normally, the FT images mentioned in the thesis are just the magnitude images. Figure 2.2 shows that the images displayed are one white rectangle in different position in the image. They both have the same FT magnitude image. The phase images would be different, of course. However, the phase image is complicated and appears less informative. In most cases of the Fourier transform for image processing and analysis, we only use the magnitude part of the FT.

If one applies the inverse transform as the definition of equation 2.2 mentioned above, figure 2.3 shows the components of the inverse transform. The elements α, β, γ are values of $F(k, l)$ and these values compose the FT images. The cosine images show the basis sine and cosine function.

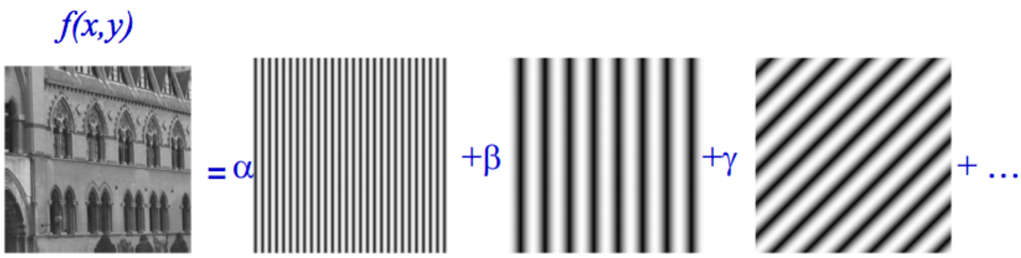


FIGURE 2.3: Inverse Fourier transform

2.2 3D visualization

Over past 15 years, 3D display technology developed very fast, it has been used everywhere in our life. Comparing with 2D space, 3D data visualization enhances the user interaction. It makes users have a totally different feeling when seeing and manipulating the object created by the same data. Thus, in this project, we will create a 3D visualization tool to show Fourier transform of image processing.

There are many 3D engines in the market (e.g., ogre3d, Unity3d ...). We choose to use the QT 3d engine in this project, because it includes a great high level 3D data visualization library(QtCompany, 2013) based on OpenGL. And it is also important that QT perfectly supports OpenCV library.

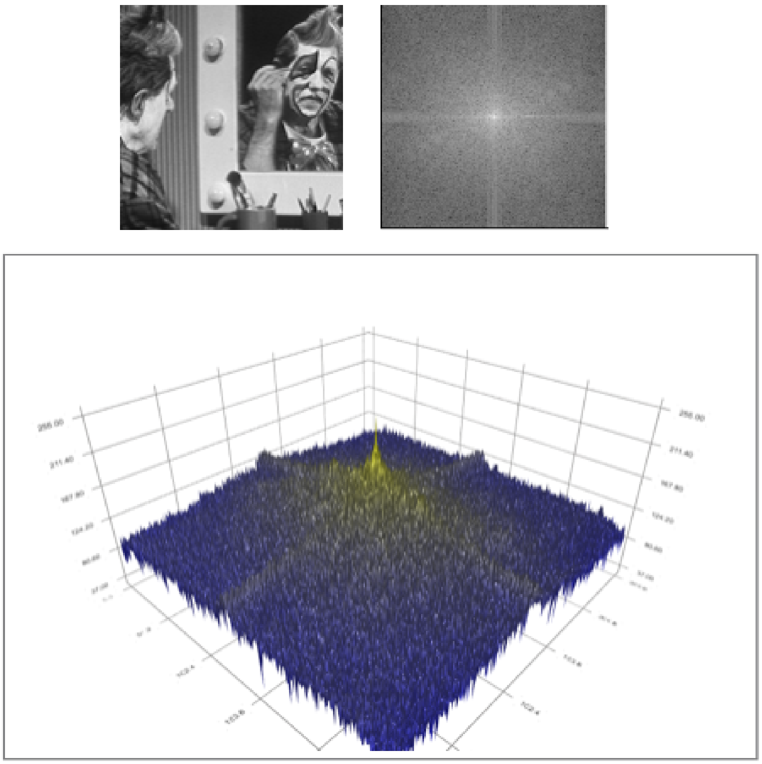
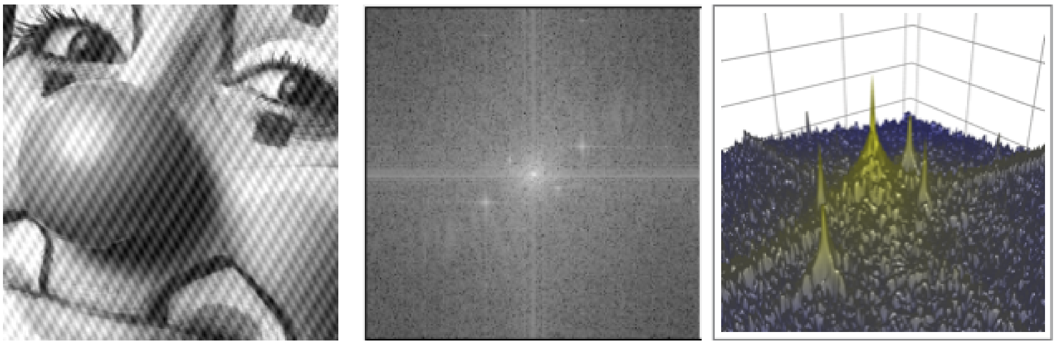


FIGURE 2.4: Example of 3D visualization for Fourier transform

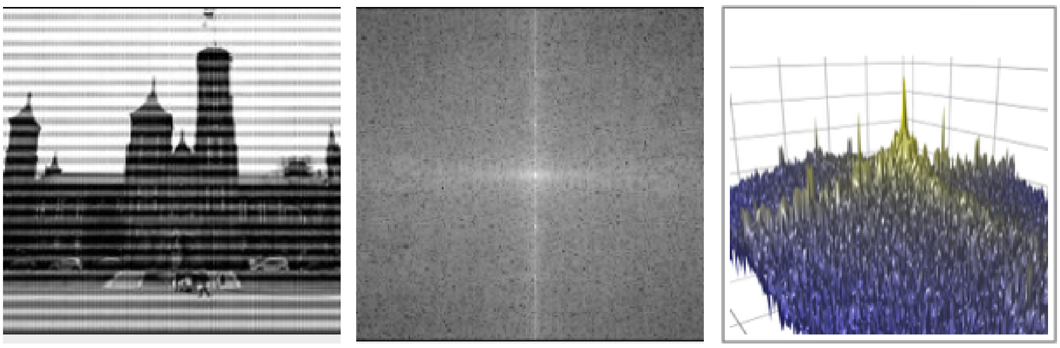
The design and implementation of the visualization are quite simple. For Fourier transform image, we use the value of pixels as the value of z-axis to create a 3D surface model. Then, this surface model is rendered in the 3D scene. Figure 2.4 is an example of 3D visualization of a Fourier transform image. As mentioned in the previous section, the center of the FT image is average value of the image, and images usually have a large average value and mainly low frequency information. Thus, in most cases the 3D surface looks a bit like a mountain, and the center is the peak.

The main function of 3D visualization is that it gives a different view of the

Fourier transform image. Figure 2.5 (A) is an example of Fourier transform of an image with bevel noise pattern. In the 2D FT image, there are four bright spots surrounding the center which represent the noise pattern. In 3D visualization, it is more obvious that there are four high peaks around the center. Another example of an image with horizontal noise pattern is presented in figure 2.5 (B). The image has a strong horizontal noise pattern, meanwhile, the image also presents a horizontal building. Thus, the 2D FT image displayed shows a white line in the y-axis and It is difficult to distinguish between the frequency points of noise pattern and the frequency of the building. In contrast, it is obvious that the peaks of the FT surface show the noise pattern of the image.



(A) 3D FT surface of an image with bevel noise pattern



(B) 3D FT surface of an image with horizontal noise pattern

FIGURE 2.5: 3D visualization of images with noise pattern

2.3 Image Filtering in frequency domain

This section will discuss image filtering in the frequency domain. The reason for doing the filtering in the frequency domain is generally because it is faster to perform two 2D Fourier transforms and a filter multiply than to perform a convolution in the spatial domain. Equation 2.3 shows the 2D convolution of a image $f(x, y)$ and a N*N filter $h(k, l)$, where $g(x, y)$ is the filtered image.

$$g(x, y) = \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} h(k, l) f(x - k, y - l) \tag{2.3}$$

Figure 2.6 represents the process of the filtering in the frequency domain. First, we compute the 2D Fourier transform of the original image $f(x, y)$. Then, by multiplying the FT images $F(u, v)$ by Filter's frequency function $H(u, v)$, we could get the filtered FT image $G(u, v)$. Finally, perform the inverse Fourier transform for the filtered FT image $G(u, v)$, the filtered image $g(x, y)$ will be obtained.

Now we will illustrate the use of some filters on the image mentioned above (Figure 2.4). The first is a lowpass filter(Bower, 2009). It removes higher frequencies, thus, the filtered image is obvious blurrier.

The ideal lowpass function shows as equation 2.4. $D(u, v)$ is a function which calculates the distance between point (u, v) to the center (0,0) of the image, and D_0 is the transition point between pass and stop bands of the filter. Figure 2.7 (A) shows the result of ideal filtering of the image mentioned above (2.4). The ideal lowpass filter completely eliminates all frequencies above the cutoff frequency while passing those below unchanged; its frequency response is a rectangular function and is a brick-wall filter(Wikipedia, 2017a).

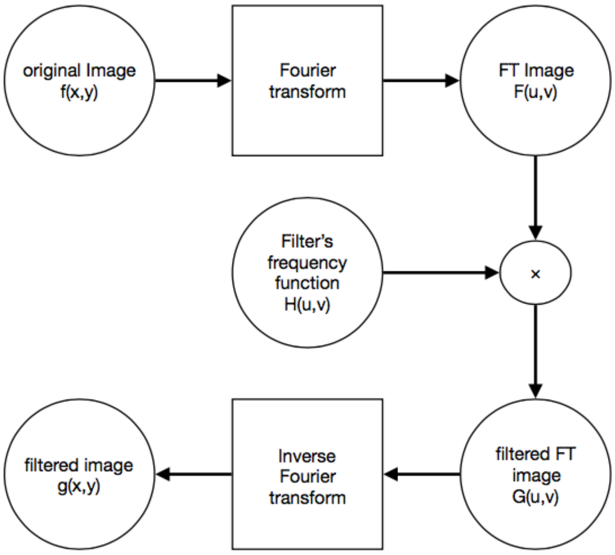


FIGURE 2.6: Workflow of image filtering in the frequency domain

The result shows that there are "ringing"[1] artifacts in the filtered imaged due to sharp cutoff of the ideal filter.

$$H_{low}(u,v) = \begin{cases} 1 & D(u,v) \leq D_0 \\ 0 & D(u,v) > D_0 \end{cases} \tag{2.4}$$

The Gaussian filter function is shown as equation 2.5. The shape is like bell-shaped curve, representing in the figure 2.7 (B) lower left. The Gaussian filtering would not cause noticeable "ringing" artifacts, because it makes a smooth transition between the pass-band and stop-band.

$$H_{low}(u,v) = e^{-(D(u,v)^2)/2\sigma^2} \tag{2.5}$$

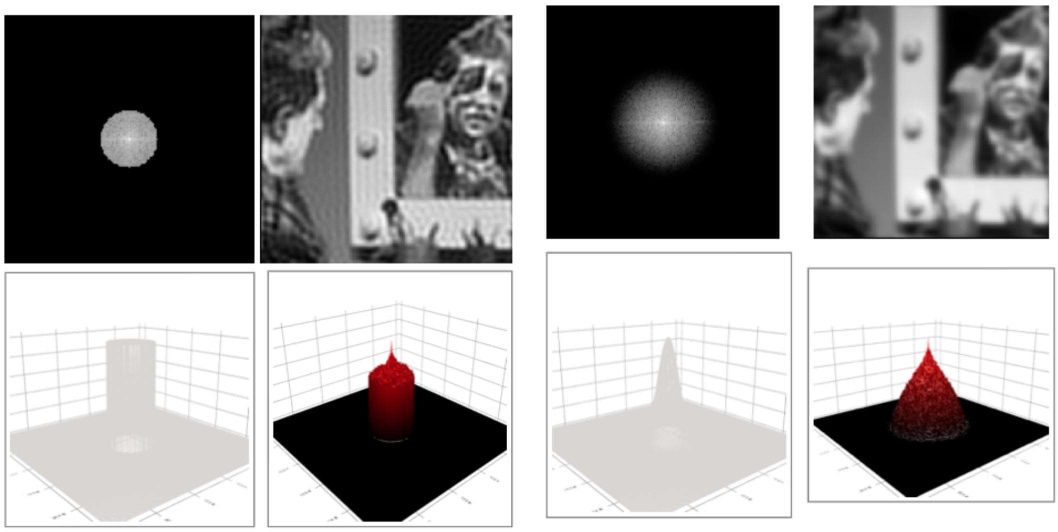
Figure 2.7 shows the ideal low pass filtering image (A) and Gaussian filtering image (B). And it also represents the 3D visualization of the filtering Fourier transform in the lower right. It shows clearly from different view in 3D that

[1] "Ringing" artifacts look like a small, light colored shadow of the edge of the object.

the ideal filter makes sharply cutoff of the FT surface and the Gaussian filter makes a smooth transition of the FT surface.

Next, the high pass filtering is to remove lower frequency content while keeping higher frequencies. The equation of high pass filtering normally is opposite to low pass filtering function as displaying in equation 2.6. The image processing of high pass filtering is to detect edges in an image, as displayed in figure 2.8. Similar as the lowpass filtering, the ideal high pass filtering would cause the "ringing" artifacts and Gaussian filter would not.

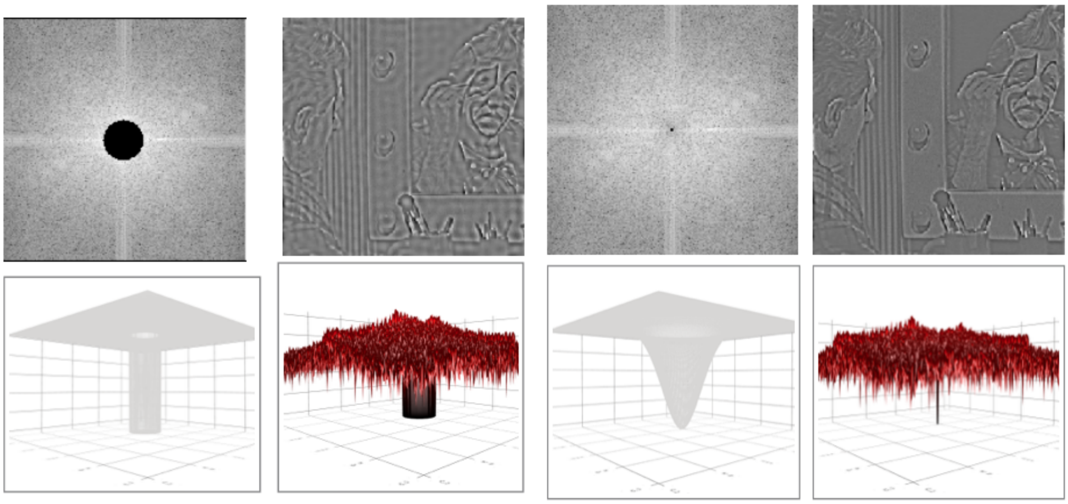
$$H_{high}(u,v) = 1 - H_{low}(u,v) \tag{2.6}$$



(A) Ideal lowpass filtering image (B) Gaussian lowpass filtering image

FIGURE 2.7: Lowpass filtering image (A is ideal filter and B is Gaussian filter); (upper left) 2D filtered FT image, (upper right) filtered real image, (lower left) 3D visualization of lowpass filter, (lower right) 3D visualization of filtered FT surface

Another application of Fourier transform is noise removal. As mentioned in previous section, the 3D visualization can help to find the noise pattern as peaks in the 3D Fourier transform surface. We use the noisy image 2.5 (A)



(A) Ideal highpass filtering image (B) Gaussian highpass filtering image

FIGURE 2.8: High pass filtering image (A is ideal filter and B is Gaussian filter); (upper left) 2D filtered FT image, (upper right) filtered real image, (lower left) 3D visualization of highpass filter, (lower right) 3D visualization of filtered FT surface

as examples. The function noise removal works like ideal high pass filtering, except that we only remove the frequency of the noise pattern but not the center point. Traditionally, as shown in figure 2.9, in 2D FT image, we will select the bright dots in the image, and create a mask filter of white background and black circles covering them. In 3D environment, we could select peaks on the 3D surface, then create a 3D mask filter (2.10).

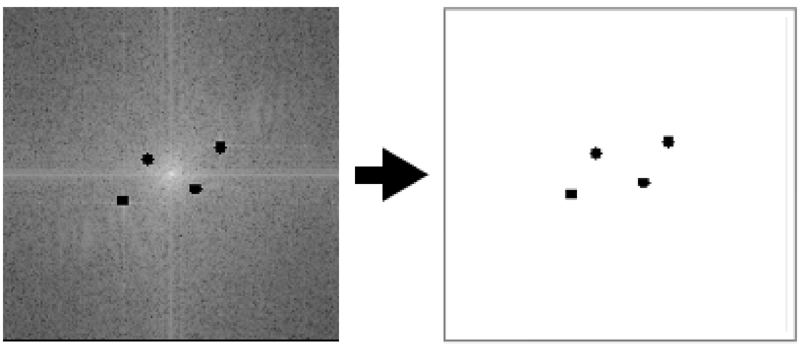


FIGURE 2.9: Custom mask filter of 2D FT image

Then, similar as filtering operation, we simply multiply the mask with the FT

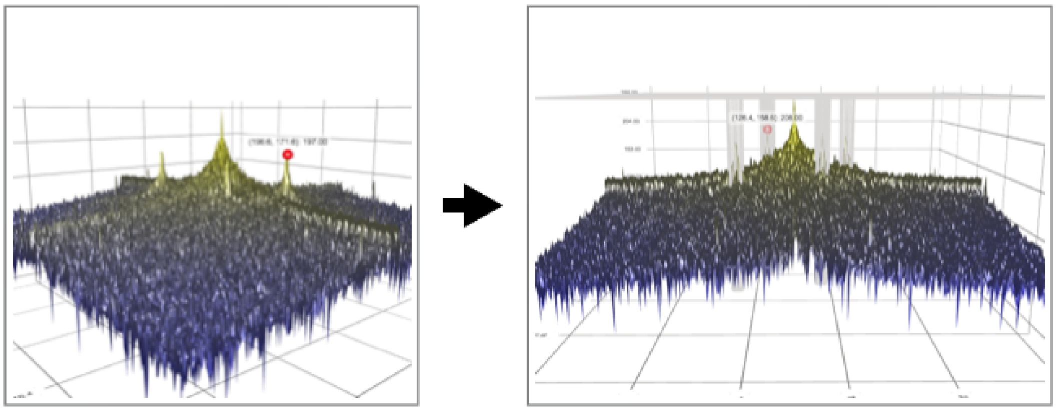


FIGURE 2.10: Custom mask filter of 3D FT surface

image, and inverse transform the filtered FT image back to spatial domain. The frequency value of noise pattern would be removed. Figure 2.11 shows the result of noise removal. In contrast, if we filtering the image by a low pass mask, we can obtain the noise pattern, as shown in figure 2.12.

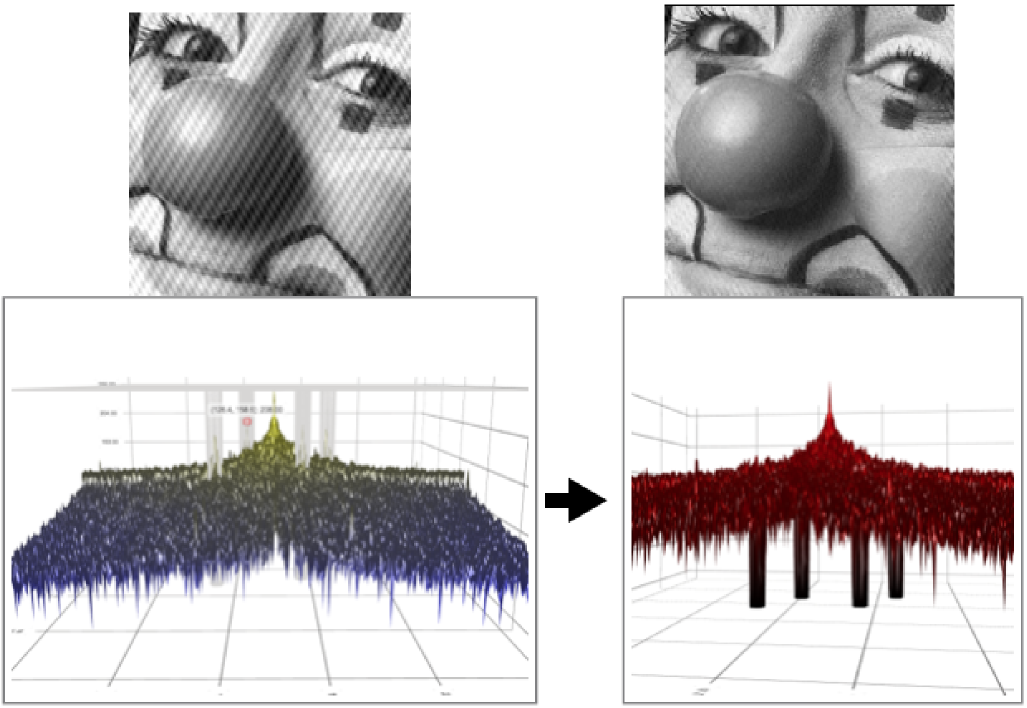


FIGURE 2.11: 3D visualization of noise removal

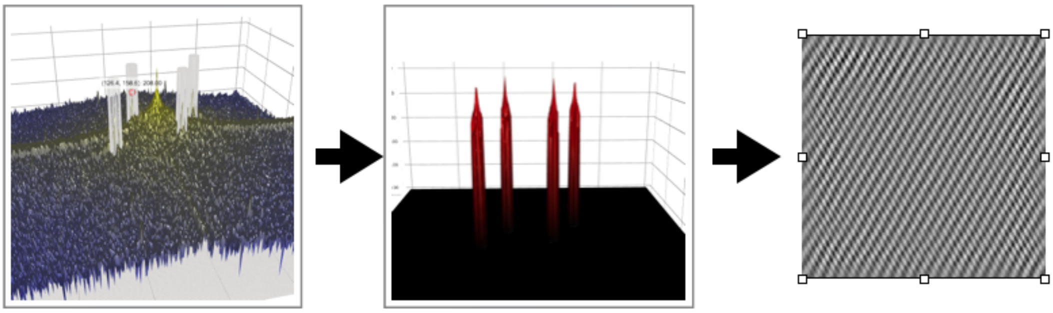


FIGURE 2.12: 3D visualization of noise pattern obtaining

2.4 Conclusion

In this chapter, we discussed the basic function of Fourier transform and 3D visualization technology applying on Fourier Transform. Then, some filtering examples were represented in 3D environment. The 3D visualization help us to understand Fourier transform principle and filtering operation easily. And it also help user to find a simple pattern of the image and create custom mask filtering to remove the noise pattern of the image.

However, the 3D visualization is limited for analyzing more complicated patterns, such as brushstrokes of paintings that we will discuss in the next chapter. Comparing with 2D image, the 3D technology represents a different world. It improves the user interaction of applications of the Fourier transform.

Chapter 3

Painting Classification using Fourier Transform

In this chapter, we will discuss the application of Fourier transform for classifying paintings. It is to distinguish paintings from different artists.

3.1 Introduction

A peculiar story to illustrate this point was in the news in 2009 (see, e.g., ["Fingerprint may lead to new da Vinci discovery" 1887]): In 2007, a gallery in New York sold a painting for 19,000 USD that at the time was thought to be a 19th century German painting. Then, two years later, art experts believed that the painting was made by Leonardo da Vinci, and suddenly, the estimated value rose to an astounding 150 million USD.

The prize level of paintings by famous artists is of course also attractive for skilled forgers; if you succeed in selling your own work as, e.g., a van Gogh painting, you will have earned enough money for a lifetime.

Traditionally, determining the authenticity of paintings have been carried out by a trained eye of human(e.g., art expert). A variety of proven techniques are used including materials dating, provenance research and morellian analysis(ArtExperts, 2017).

Sometimes it is possible to make an absolute classification of a painting. However, with the ever growing number of forgeries, it is becoming more difficult to distinguish the authentic works by merely using human experts. In any case, when the tasks of authenticating paintings is performed by art experts the classification process is a costly and a subjective procedure. Fortunately, based on the differences among painters in terms of canvas textures, pigments, and brushstroke styles, it has been discovered that the mathematical analysis of the digital representation of paintings could assist the art experts in the authentication work. Recently, the painting identification by digital image processing techniques has become popular and various approaches have been proposed.

In some cases, image analysis based on the spatial domain (e.g., color analysis, wavelet analysis) is used, and we will introduce some methods with this approach in the next section. In contrast, painting analysis in the frequency domain did not develop fast and lacks methodology. Fourier transform is a powerful tool of image analysis, and we will also give some related work in the next section. In this project, we will use Fourier transform of image processing to classify the paintings from a group of artists.

3.2 Related Work

In this section, we will give a quick introduction of the methods for painting analysis and Fourier transform from other papers.

3.2.1 Painting analysis

Visual characteristics of paintings such as color, brushwork, and composition concepts constitute a large body of expert analysis in the paintings domain(Arnheim, 1954). These features tightly relate to high-level semantic information of painting such as artist name, painting styles and periods int art history. Thus, metrics of these features have been used for painting analysis to support applications such as brushstrokes detection, painting authentication and anti-fakery analysis.

Color Analysis

A method used in the analysis of painting based on color is called "Palette"(Rigau et al., 2010). Figure 3.1 represents an example using this type of technique to show the changes in the artworks by Van Gogh in his different periods. Thomas Lombardi from Pace University created a palette description algorithm based on an HSV color model to extract features(Lombardi, 2005) and use these features (including other color features such as Hue Histogram, Saturation Histogram and RGB Histogram) to perform painting classification. The experiment use a dataset which contains ten paintings each from the work of Cezanne, Monet, Pissarro, Seurat, Sisley, and Van Gogh. And the classification results of each painter are 100%, 20%, 0, 60%, 20%, 0 respectively.

Brushstroke Analysis

Brushwork is important for the annotation of artistic concepts in the paintings domain. Thus, analyzing the brushstroke is valuable and effective for the annotation of paintings with artist names. Basically, the brushstroke is similar to a texture image (figure 3.4 is an example that shows the texture of brushstroke style of a painting), and several studies performed brushstroke

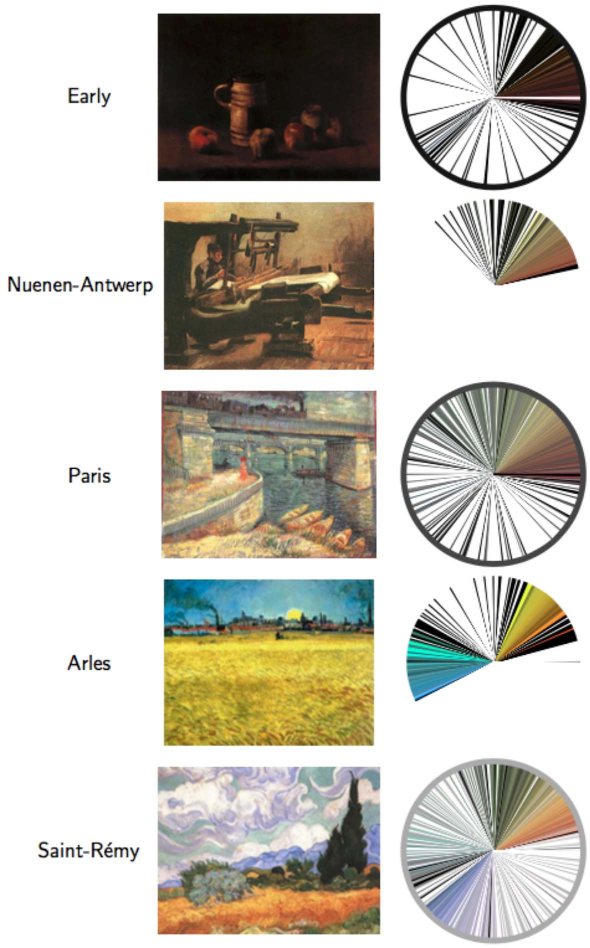


FIGURE 3.1: Digital-image-palette of van Gogh's periods(Rigau et al., 2010).

analysis and utilized it for painting classification.

One method based on a segmentation algorithm is introduced by C. Richard Johnson(Johnson, Hendriks, and Berezhnoy, 2008). By applying a wavelet filter on the digital painting image and extracting the brushstroke contour lines(3.2), followed by comparing the feature vectors between different paintings, a similarity between the artists could be shown. The test dataset including 101paintings which 82 have consistently been attributed to van Gogh, six have always been known to be non-van Gogh, and others are currently questioned by experts. The result of the classification is that four out of the six non-van Gogh paintings were detected. Ella Hendriks also used a similar

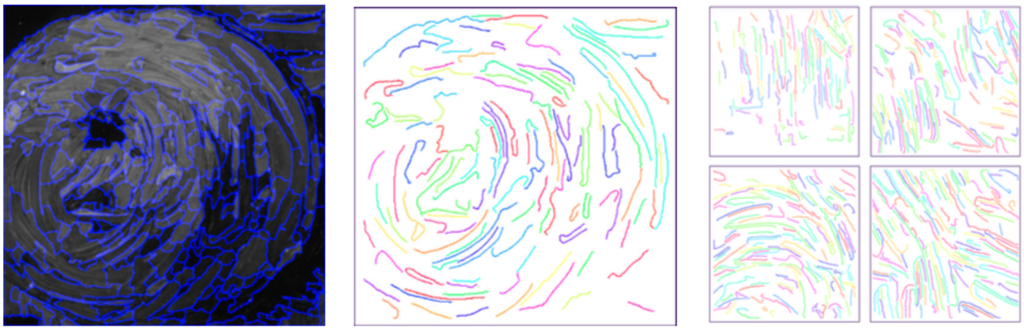


FIGURE 3.2: Wavelet analysis example (Johnson, Hendriks, and Berezhnoy, 2008)

wavelet analysis method to provide authenticity of van Gogh's works(Hendriks and Hughes, 2007).

Another highly complex method was introduced by Jia.Li (Li, Yao, and Hendriks, 2011). Basically, using a segmentation algorithm and an edge detection algorithm, brushstroke backbones can be automatically detected(3.3). Afterwards, this method extracts the following features.

1. Number of Brushstrokes in the Neighborhood(NBS-NB).
2. Number of Brushstrokes with Similar Orientations in the Neighborhood (NBS-SO)
3. Elongatedness
4. Straightness
5. Broadness Homogeneity (BH)

The experiment uses forty-five paintings as a dataset which includes thirty-one van Gogh paintings versus fourteen paintings by others. The accuracy of classification between van Gogh and other painters from these features, the values are 85.7% for NBS-NB, 78.6% for NBS-SO, 85.7% for Elongatedness, 85.7% for Straightness, and 87.10% for BH respectively.

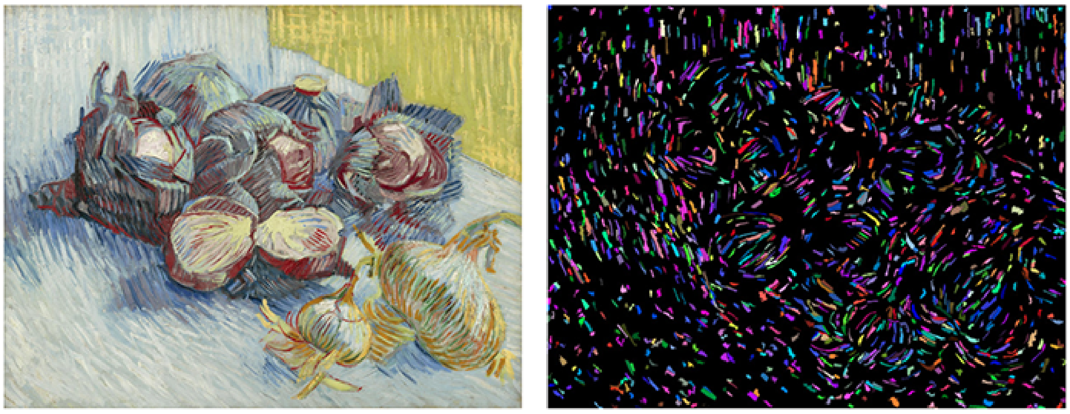


FIGURE 3.3: Brushstroke extraction results(Li, Yao, and Hendriks, 2011)

The solutions mentioned above give an framework and image processing methods to extract features from the brushwork texture. The results are encouraging. However, they are limited because the analysis only focuses on the artwork of van Gogh, and needs to extend the analysis to the other artists.

3.2.2 Pattern analysis using Fourier transform

Fourier transform magnitudes are commonly used in pattern recognition applications. The analysis of Fourier transform magnitudes could demonstrate robustness and rotation of the pattern of the image. Several studies performed Fourier transform analysis and apply it to pattern recognition.

One method from Ahsan Ahmad Ursani(Ahsan Ahmad Ursani, 2008) uses a texture descriptor called Local Fourier Histograms(LFH) to recognize the texture image. It uses a window DFT algorithm to calculate the coefficients of textures. The accuracy of recognition of the texture image out of 2560 images is about 90%.

Another study of Fourier analysis for Palmprint Recognition is from LI Wenxin(LI Wen-xin, 2002). They introduce a polar coordination system (r, θ) based on the Fourier transform to represent the palmprint features. The experiment includes a dataset of 500 images of palmprint, and its accuracy is 95%.

Both studies shows that Fourier transform is a powerful tool to do the pattern analysis. The accuracy of the pattern recognition is high. It is a great solution for solving the problem of analyzing texture images with patterns.

3.2.3 Summary

As mentioned before, painting classification has either low accuracy or is limited to one specific painter(e.g., van Gogh). The Fourier transform instead are highly accurate for pattern recognition of texture image. And the brushstroke image is one type of texture image. Thus, it is possible to apply Fourier transform on brushstroke analysis and improve the accuracy of painting classification.

In this project, we will design and create a method based on Fourier analysis, and use this method to perform painting classification with a multi-class dataset.

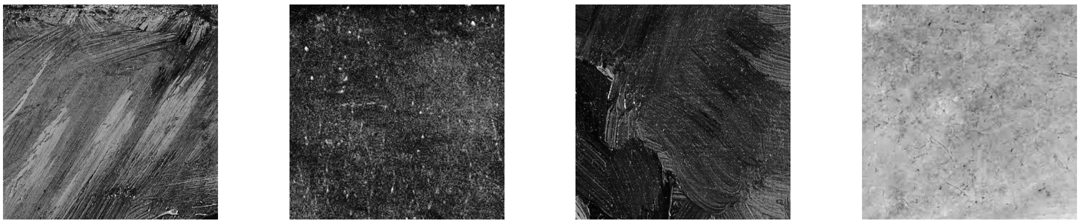
3.3 Theoretical Background

In this section, we will introduce the basic knowledge about brushstroke analysis using Fourier transform and a quick view of a machine learning algorithm, the support vector machine.

3.3.1 Brushstroke analysis using Fourier Transform

Fourier Transforms have been introduced in chapter 2, and in this part, we will introduce how to perform brushstroke analysis by Fourier transform.

Brushwork serves as an important cue to the identification of artist. Figure 3.4 shows some different kinds of brushstroke texture image. The patch 1 includes directional edges and with high contrast and patch 3 also contains edges but with lower contrast. Patches 2 and 4 show an homogeneous texture.



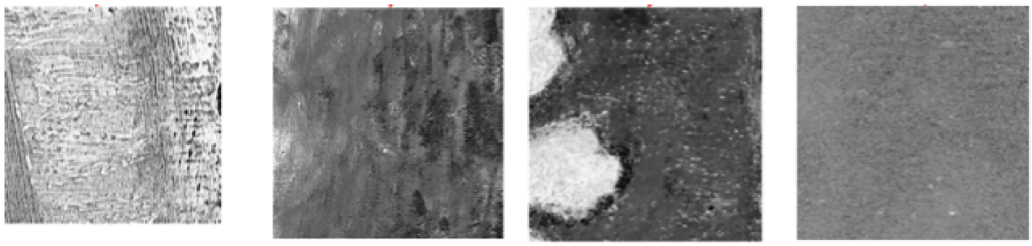
(A) Patch 1 (B) Patch 2 (C) Patch 3 (D) Patch 4

FIGURE 3.4: Texture patches of a painting shows the brushstrokes information

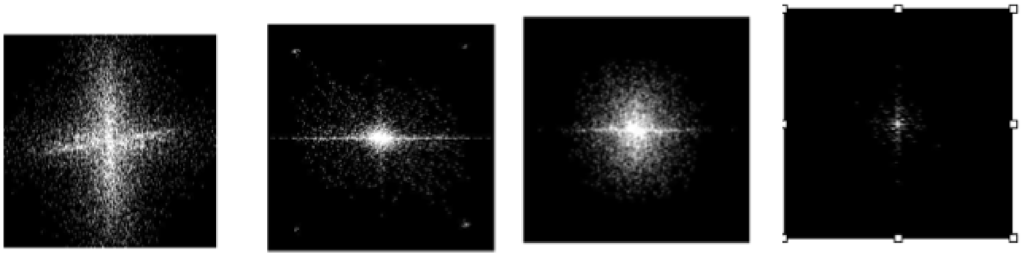
As mentioned in previous section, Fourier transform magnitudes are robust and show the rotation of the pattern of the image. We perform the Fourier transform for the brushwork texture image and remove the low value of the frequency, because the high value of the Fourier transform image represents the pattern features. Figure 3.5 (A) shows four different type of the FT images of the brushstroke textures. A1 is a texture with horizontal and vertical edge; A2 shows only one directional edges; A3 is a smooth texture; A4 is homogeneous texture. The FT images displayed in Figure 3.5 (B) shows some attributes as below.

1. The size of the FT spectrum represents the strength of brushstroke.
2. The rotation of the FT spectrum represents the direction of brushstroke.

Based on these two attributes, we choose the size of FT spectrum to represent the feature of the brushstroke texture, because the direction of the brushstroke mostly is decided by the contents of the paintings. When the artist



(A) Different type of brushstroke texture image 1-4



(B) The FT image of the texture keeping the high value 1-4

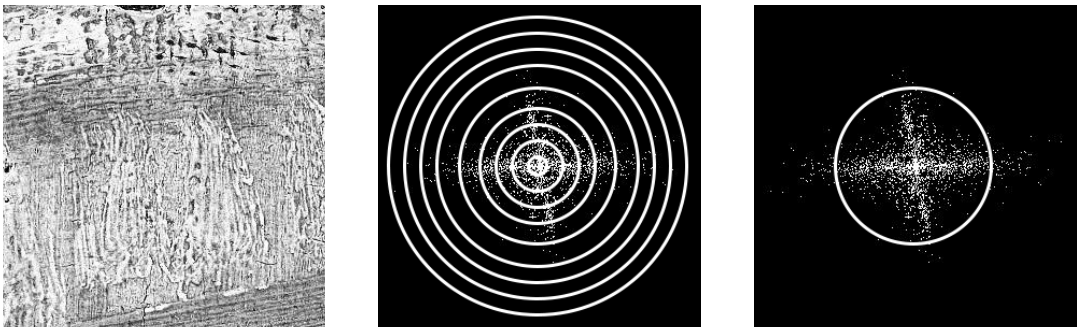
FIGURE 3.5: Fourier transforms of different type of brushstroke textures

paints different things or different part of an object, the direction will be different. That is the reason one painting could include different directional brushstrokes.

The intensity of the Fourier spectrum is sufficient to show the strength of brushstroke, and the spectrum size is calculated as the strength of brushstroke. The algorithm 1 shows the process of computing the Fourier spectrum size. First, we use the brushstroke texture image and the max size of FT(e.g., 10) as the input. The FT result of the image is computed, and a threshold is applied for removing the low value of the frequency. The result looks like the FT images shown in the figure 3.5 (B). Then, we draw a black circular mask from maximum size to minimum size on the FT image . After drawing the mask on the FT image, we count the sum of the bright dots of the FT image. When a small quantity of bright dots are out of the mask, this mask size will be returned as a result which shows the size of the Fourier spectrum. Figure 3.6 shows an example of calculating the size of Fourier spectrum.

Algorithm 1 Compute Spectrum Size

```
function COMPUTESPECSIZE(image, max)
    complex ← DFT(image)
    spec ← COMPUTEMAG(complex)
    center ← POINT(spec.cols/2, spec.rows/2)
    radius ← spec.cols/2
    for i = 0 TO max do
        r ← radius − radius ∗ (i/level)
        mask ← CIRCLE(center, r, black)
        //draw mask on the FT image
        specfiltered ← spec + mask
        //counting the sum of bright dots of the FT image
        rate ← SUM(specfiltered)/SUM(spec)
        if rate > 0.08 then return (max - i +1)
```



(A) (B) (C)

FIGURE 3.6: An example of calculating the intensity (A) Patches from painting (B) different masks of the Power spectrum (should be black) (C) Size of the Power spectrum

3.3.2 Support vector machine

The machine learning algorithm is not main topic of this thesis. Thus, we will only give a quick overview of it.

Classifying data is a common task in machine learning. Suppose some given data points belonging to one of two classes, and the goal is to decide which class a new data point will be in(Wikipedia, 2017b). In this project, we use the support vector machine(SVM) algorithm implemented in OpenCV(OpenCV, 2017b).

The SVM algorithm is a discriminative classifier that is formally defined by a separating hyperplane. The SVM model requires labeled training data for initialization and outputs an optimal hyperplane that categorizes new examples(3.7).

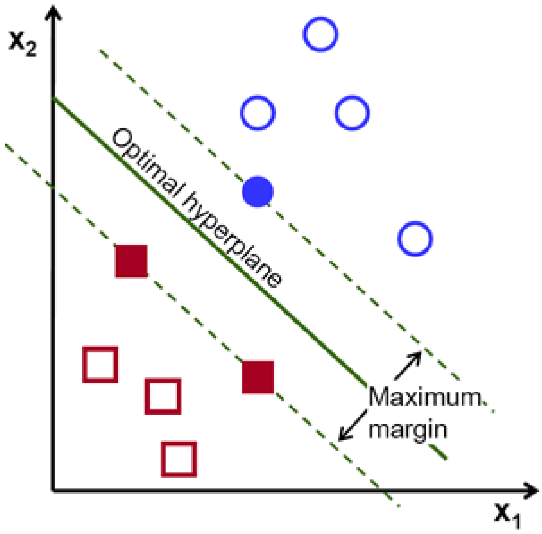


FIGURE 3.7: Example of Support Vector Machine from OpenCV(OpenCV, 2017b)

3.4 Proposed Approach

In this section, we will describe more accurately and in more detailed about the classification model. In addition, we will give some examples of each step. The workflow of the process is shown in figure 3.8.

The process consists of several parts.

1. First, histogram equalization is applied to the digital painting image to enhance the brightness and contrast and obtain more brushstroke details.

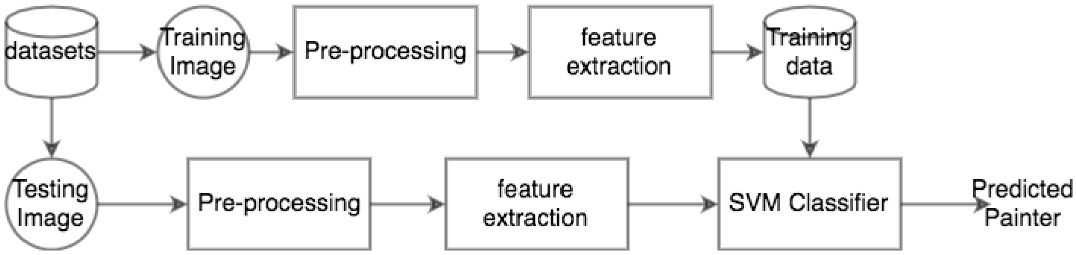


FIGURE 3.8: Proposed approach work flow

2. Then the large image is cropped to many small patches because only a small patch shows the texture of the brushstroke.

3. A discrete Fourier transformation (DFT) is applied to every patch to extract spectrum size and generates the feature matrix.

4. From this spectrum feature matrix, we count the number of patches by different strength and calculate the histogram feature vector.

5. For creating the support vector machine (SVM), whole dataset will be divided into training set and testing set. We calculate the vectors from all of the training sets and input them to the SVM model to perform the initialization.

6. Finally, this SVM classifier predicts the painter of the painting from the testing dataset.

In the next sections, we will use the painting "Exterior of a Restaurant in Asnieres"(3.9) from Vincent van Gogh(Gogh, 2009) as an example to show the result of each step of the working flow.

3.4.1 Painting analysis with Fourier transform

First, painting images collected from the internet vary in brightness and contrast, and thus, the content of a brushstroke is totally different even from the same paintings. Thus, in the preprocessing step, the image must be adjusted to a state which is almost the same brightness and contrast in order to make

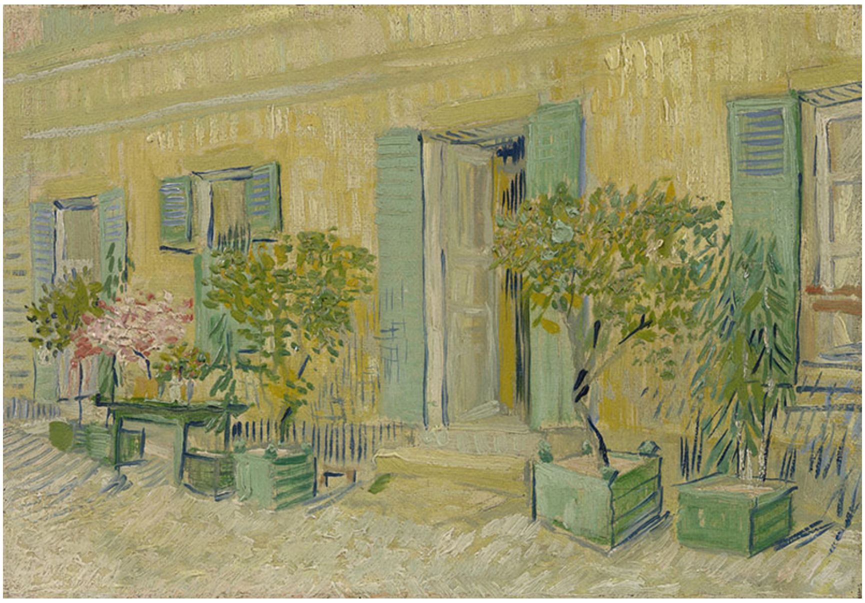


FIGURE 3.9: "Exterior of a Restaurant in Asnieres" by Vincent van Gogh(Gogh, 2009)

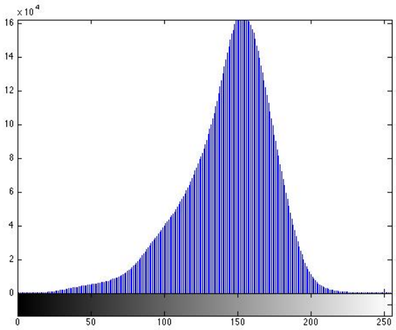
the Fourier analysis more accurate in the next step. The results of preprocessing and comparison are shown in figure 3.10. The processing uses the opencv library(OpenCV, 2017a).

As mentioned above, the Fourier transform can extract the size of Fourier spectrum. Similar brushstrokes will obtain similar size of the Fourier spectrum. Based on this idea, the strength of brushstrokes from the same painter could be more similar than another painter. In order to obtain the brushstroke information, we divided one large high resolution painting image into many small patches with the brushstroke texture (3.11).

Recall the previous algorithm 1, the size of Fourier spectrum for each brushstroke texture image is calculated, and all these numbers consist of a matrix that shows the feature of the strength of brushstroke texture of the whole painting (Figure 3.12 shows an example of the brushstroke matrix).

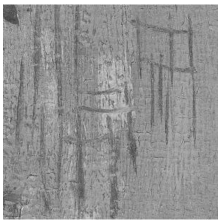
The matrices between two paintings are difficult to compare, because the

(A) B/W image without processing

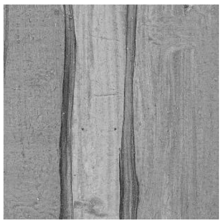


(B) Histogram of the image

(C) Patch 1



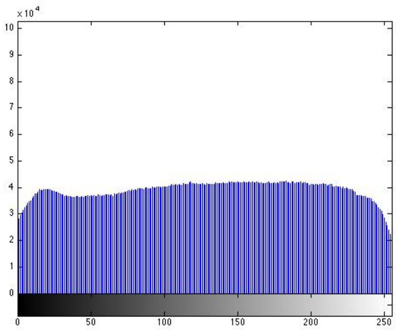
(D) Patch 2



(E) Patch 3

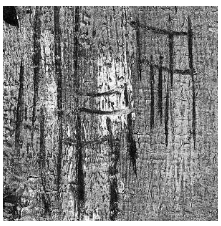
(F) Patch 4

(G) B/W image after processing

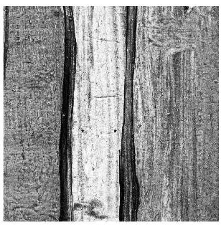


(H) Histogram after processing

(I) Patch 1



(J) Patch 2



(K) Patch 3

(L) Patch 4

FIGURE 3.10: The output of the preprocessing and comparison (A)-(F) shows the results before histogram equalization, while (G)-(L) show the results after processing

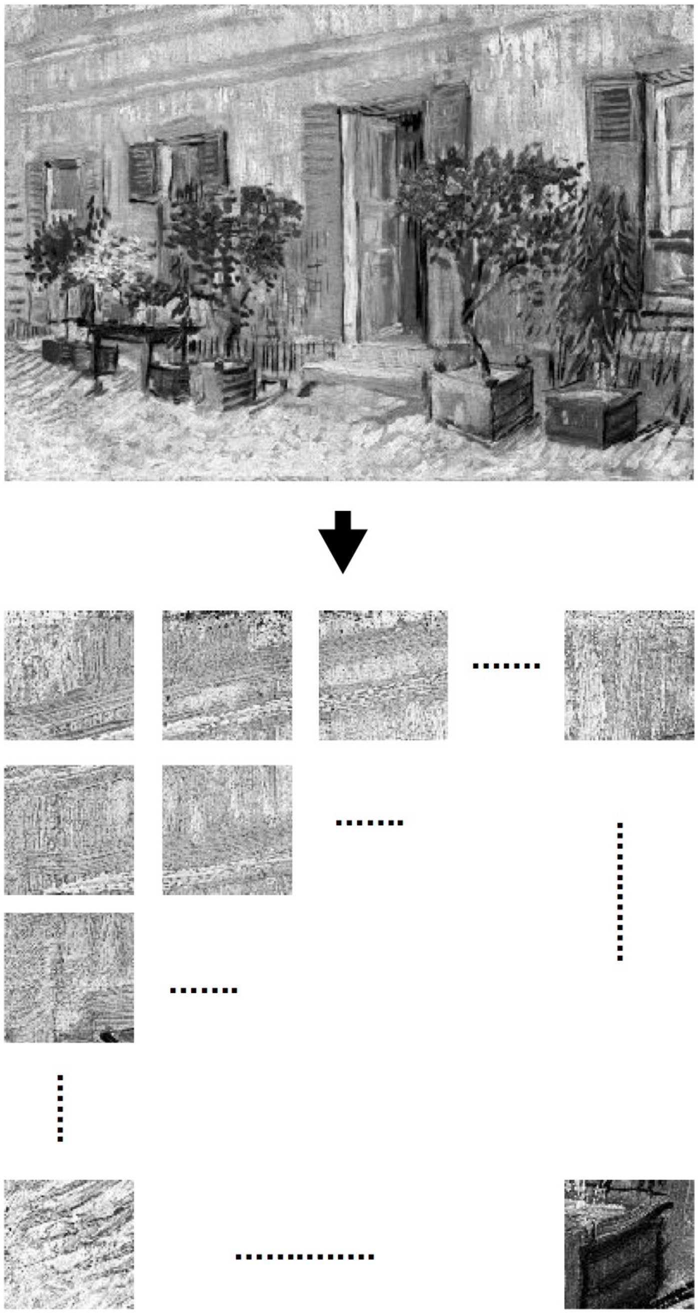


FIGURE 3.11: Example of cropping: dividing the painting into many small patches, including the brushstroke information.

contents of the painting decided the matrix content, e.g., it is totally different between paintings of still life and paintings of landscape (Figure 3.15 shows an example of two brushstroke matrices from two paintings created by Van Gogh). Thus, in the next step, the matrix has to be converted to a structure independently. We used the counting histogram algorithm(2) to calculate the

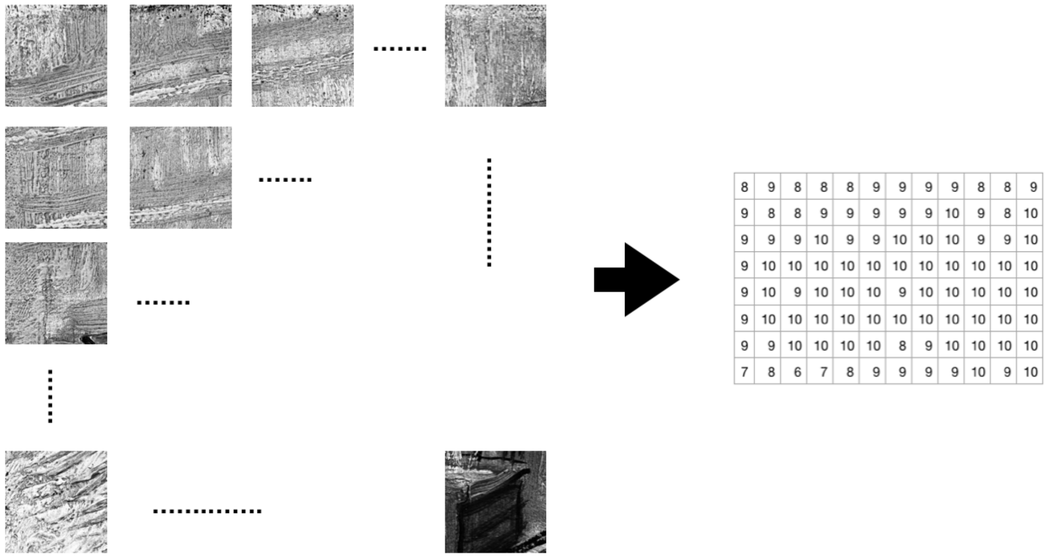


FIGURE 3.12: An example of the brushstroke feature matrix.

histogram vector(Figure 3.13 shows an example). The histogram shows the distribution of the brushstroke strength.

Algorithm 2 Compute Feature Histogram

```
1: function COMPUTEFEATURE(matrix, vectorSize)
2:     vec_mat ← CONVERT2VECTOR(matrix)
3:     vec_mat ← SORT(vec_mat)
4:     vec_output ← INIT(0, vectorSize)
5:     j ← 0
6:     for i = 0 TO size(vecmat) do
7:         vec_output[j] ← COUNTING(vec_mat, j + 1)
8:         j ← j + 1
9:     vec_output ← NORMALIZE(vec_output) return vec_output
```

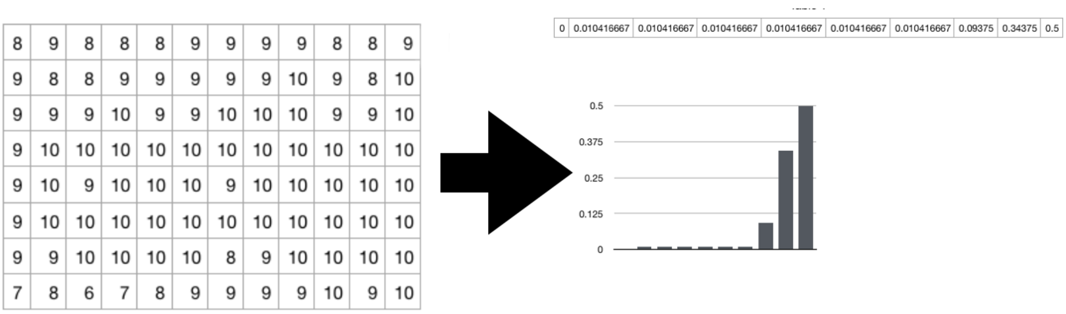
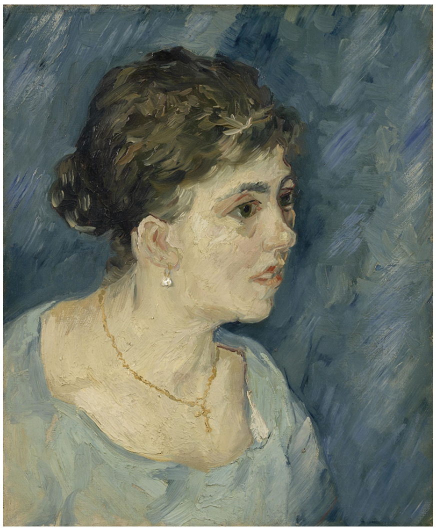


FIGURE 3.13: An example of the brushstroke feature histogram.

(A)

8	9	8	8	8	9	9	9	9	8	8	9
9	8	8	9	9	9	9	9	10	9	8	10
9	9	9	10	9	9	10	10	10	9	9	10
9	10	10	10	10	10	10	10	10	10	10	10
9	10	9	10	10	10	9	10	10	10	10	10
9	10	10	10	10	10	10	10	10	10	10	10
9	9	10	10	10	10	8	9	10	10	10	10
7	8	6	7	8	9	9	9	9	10	9	10

(B)



(C)

11	10	10	9	9	10	10	10
10	10	11	10	11	10	9	9
9	10	11	10	10	10	10	8
10	10	11	9	6	7	9	8
10	10	9	4	4	6	9	6
10	9	6	3	4	7	9	7
7	5	2	3	6	7	7	8
4	2	2	2	4	7	9	9
4	4	2	3	2	5	9	6
4	4	4	5	4	2	8	8

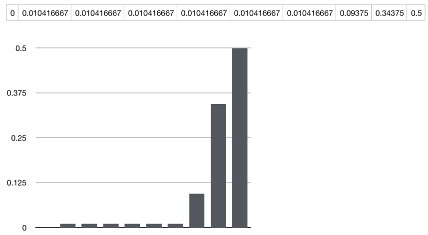
(D)

FIGURE 3.14: The differences in the matrixes from two paintings created by Van Gogh.

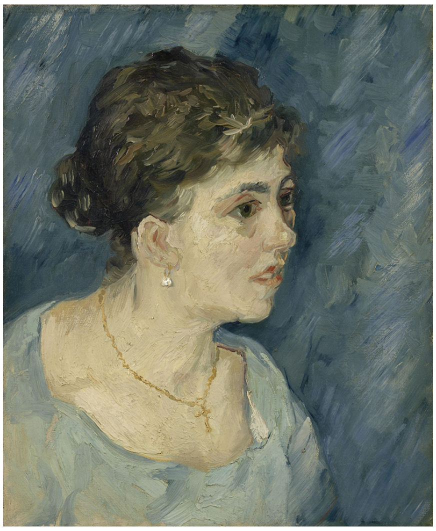
Next, the features of a histogram vector are shown in the context of brushstroke strength, and it is not relative to the content of the painting. Figure 3.15 shows the histogram of the paintings above. It shows that the distribution of brushstroke strength of two paintings is similar.

Using such multidimensional histogram representations of paintings, the painting classification framework induces a model by performing automatic classification experiments with a support vector machine(SVM, mentioned above). The SVM requires a training set with input vectors and labels of their class. In this project, the input vectors are the 10-dimensional-vector histograms introduced in this section; the labels represent the authorship of the painting. The SVM learns to create a mapping relationship from the input vectors to

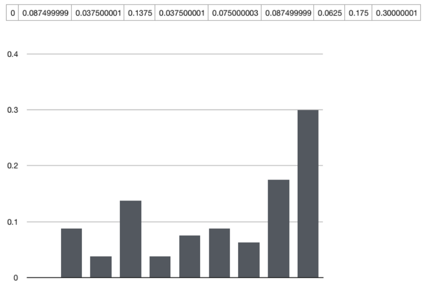
(A)



(B)



(C)



(D)

FIGURE 3.15: The differences in the histograms from two paintings created by van Gogh.

the labels by the training set. After training, the SVM could be used to predict the authorship of the test painting from its feature vector.

3.4.2 Parameters of feature extraction

In the process of feature extraction, several parameters could affect the result. These parameters currently are deceided by the user.

1. Patch size: it is the size of patch.

2. Feature vector size: It defines how large the histogram will be, decided by spectrum max size.

3. spectrum max size: mentioned in previous section, the algorithm 1 uses cycles to define the spectrum size, this parameter is the max size of the cycles. It also decides the histogram size.

4. Fourier spectrum threshold: In the algorithm 1, the spectrum image is applying a threshold which remove the low value of the frequency.

The parameters will affect the last result of the feature extraction. However, it is difficult to decide which value is perfect for obtaining the best results. We heuristically tried out many different settings for the parameters, and the values (shown below) we used in the project could be shown as a acceptable solution. In the next section, we will use this parameter configuration in the experiments.

The current configuration of parameters is:

1. Patch size: 300px * 300px

2. Feature vector size: 10

3. Spectrum max size: 10

4. Fourier spectrum threshold: 0.6

3.5 Experiment Results

In this section, we will present the details of the experiments on paintings related to several artists. All of the experiments were conducted in Qt with OpenCV library, and the source code is distributed[1].

3.5.1 Datasets and Configuration

We have worked with paintings related to 6 different artists: Gauguin, Monet, Rembrandt, Rubens, Toulouse Lautrec and van Gogh. The images should be high-resolution images(> 3 MP). We downloaded them from Wikimedia[2].

For each artist, the dataset contains 30 paintings. Half of them are used as training data, and the remainder are the testing data. The artwork is listed in Appendix A , and figure 3.19 shows some examples.

As mentioned above, for the experiments, we used one type of parameters for feature extraction, as follows.

1. Patch size: 300px * 300px
2. Feature vector size: 10
3. Spectrum max size: 10
4. Fourier spectrum threshold: 0.6

Comparisons with different parameters will not be investigated thoroughly here, and these tasks need further investigation and will be addressed with in our subsequent work.

[1] `https://github.com/gegego/paclas`
[2] `https://commons.wikimedia.org/`

(A) Gauguin

(B) Monet

(C) Rembrandt

(D) Rubens

(E) Toulouse Lautrec

(F) Van Gogh

FIGURE 3.16: Some examples from datasets that were downloaded from Wikimedia[2]

3.5.2 Results and Analysis

This section will present the results obtained by the 2-class classification and other multi-class tests.

Table 3.1 shows the results of 2-painter classification. The accuracy of most tests is higher than 80%. However, the results of three tests (Toulouse Lautrec vs Gauguin, Gauguin vs Rubens and Rembrandt vs Rubens) are lower than the others: 60%, 60% and 73%, respectively.

Next, we calculate the distance between these painters. The algorithm computes the distance of each pair of paintings by the formula 3.1 and takes the

Authors	Accuracy
Van Gogh / Monet	96.6667
Van Gogh / Toulouse Lautrec	100
Van Gogh / Gauguin	96.6667
Van Gogh / Rembrandt	100
Van Gogh / Rubens	100
Monet / Toulouse Lautrec	100
Monet / Gauguin	93.3333
Monet / Rembrandt	90
Monet / Rubens	90
Toulouse Lautrec / Gauguin	60
Toulouse Lautrec / Rembrandt	86.6667
Toulouse Lautrec / Rubens	83.3333
Gauguin / Rembrandt	80
Gauguin / Rubens	60
Rembrandt / Rubens	73.3333

TABLE 3.1: Results of 2-class experiments.

average value, as illustrated in Algorithm 3. The results are shown in table 3.2. The distance between the above three pairs of painters, which are low in accuracy, is lower – 0.0890975, 0.091223 and 0.093803, respectively – and the others are all larger than 0.1.

$$D = \frac{1}{N} \sum_{k=0}^{N-1} |X_i - Y_i| \tag{3.1}$$

Algorithm 3 Compute Average Distance

function COMPUTEAVGDIST(asets, bsets)
    $d \leftarrow 0$
    for `i = 0 TO asets.size()` **do**
        for `j = 0 TO bsets.size()` **do**
            $d \leftarrow d + \text{COMPUTEDISTANCE}(asets[i], bsets[j])$
    return d/(asets.size()*bsets.size())

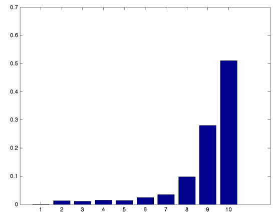
And next, we will visualize and analyze the data. Figure 3.17 shows the bar

Authors	Average Distance
Van Gogh / Monet	0.151286
Van Gogh / Toulouse Lautrec	0.184208
Van Gogh / Gauguin	0.175351
Van Gogh / Rembrandt	0.167368
Van Gogh / Rubens	0.166563
Monet / Toulouse Lautrec	0.15936
Monet / Gauguin	0.140883
Monet / Rembrandt	0.125144
Monet / Rubens	0.123739
Toulouse Lautrec / Gauguin	0.0890975
Toulouse Lautrec / Rembrandt	0.114944
Toulouse Lautrec / Rubens	0.10621
Gauguin / Rembrandt	0.104255
Gauguin / Rubens	0.091223
Rembrandt / Rubens	0.093803

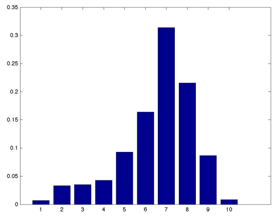
TABLE 3.2: Average Distance between Artists

charts of the mean value of the feature histograms. (C),(D) and (F) have similar brushstroke strength differing from the others.

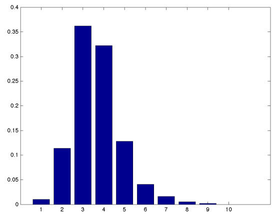
Then, we use a a parallel coordinate plot(Matlab, 2006) to show the result. Parallel coordinates is a visualization technique used to plot individual data elements across many dimensions. Each of the dimensions corresponds to a vertical axis and each data element is displayed as a series of connected points along the dimensions/axes(Gemignani, 2010). As illustrated in figure 3.18 (A), each line shows one painting of the artist. The blue line represents artwork of van Gogh and green line represents Monet. The lines show that the brushstroke of van Gogh is concentrated in the strength of 10 and Monet's brushstroke is concentrated in the strength of 7. Figure 3.18 (B) is parallel coordinates plot with quantile value of 0.25. The quantile is a numeric value in the range (0,1). If a value α for 'Quantile' is specified, then parallel coordinate plots only the median, α, and $1 - \alpha$ quantiles for each of the attributes(axes). The quantile plot option provides a useful summary of the data when dataset contains many observations. Parallel coordinate plot



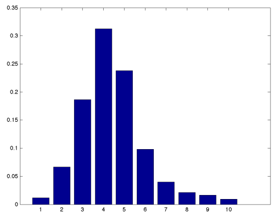
(A) Van Gogh



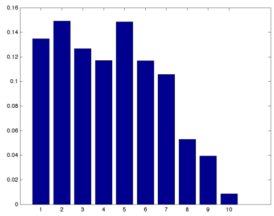
(B) Monet



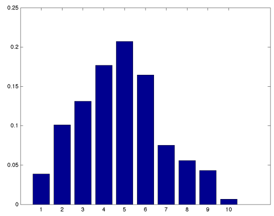
(C) Toulouse Lautrec



(D) Gauguin



(E) Rembrandt



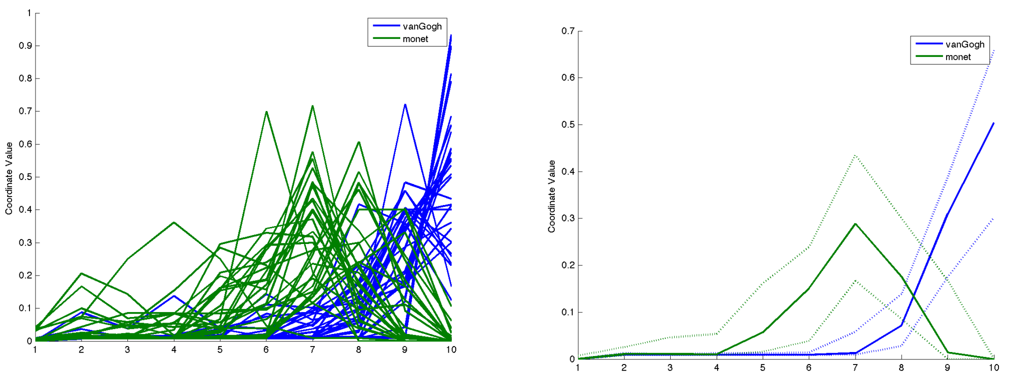
(F) Rubens

FIGURE 3.17: Bar Chart of the mean values of all of the features for each painter.

used in the project is implemented in Matlab(Matlab, 2006).

Figure 3.19 shows parallel coordinates plot with quantile value of all of the data in 2-class experiments. The difference of brushstroke strength between painters is obvious. As shown in previous table, the results of three experiments (Toulouse Lautrec vs Gauguin, Gauguin vs Rubens and Rembrandt vs Rubens) are low. The reason is shown in the corresponding figures, (J), (N) and (O) show the distribution of brushstroke strength of these three pairs of painters. The brushstroke of these three painters are all concentrated in the strength of 4, and the distributions of the other brushstroke strength are also similar. The other figures show that distribution of brushstroke strength is different between painters and this difference makes the high accuracy of the classification.

The similarity of brushstroke strength of the paintings from painters of these three pairs induces low accuracy in the experiments, and it also affects the



(A) normal parallel coordinate plot of van Gogh and Monet

(B) Quantile plot of van Gogh and Monet

FIGURE 3.18: Parallel coordinate examples

results of other multi-class experiments. The results of the multi-class experiments are shown in table 3.3, 3.4, 3.5 and 3.6. The test cases including the above three pairs of painters also obtain a low accuracy of the classification.

Authors	Accuracy
Van Gogh / Monet / Toulouse Lautrec	97.7778
Van Gogh / Monet / Gauguin	93.3333
Van Gogh / Monet / Rembrandt	91.1111
Van Gogh / Monet / Rubens	91.1111
Van Gogh / Toulouse Lautrec / Gauguin	71.1111
Van Gogh / Toulouse Lautrec / Rembrandt	91.1111
Van Gogh / Toulouse Lautrec / Rubens	88.8889
Van Gogh / Gauguin / Rembrandt	86.6667
Van Gogh / Gauguin / Rubens	73.3333
Van Gogh / Rembrandt / Rubens	82.2222
Monet / Toulouse Lautrec / Gauguin	68.8889
Monet / Toulouse Lautrec / Rembrandt	84.4444
Monet / Toulouse Lautrec / Rubens	82.2222
Monet / Gauguin / Rembrandt	80
Monet / Gauguin / Rubens	66.6667
Monet / Rembrandt / Rubens	73.3333
Toulouse Lautrec / Gauguin / Rembrandt	57.7778
Toulouse Lautrec / Gauguin / Rubens	48.8889
Toulouse Lautrec / Rembrandt / Rubens	75.5556
Gauguin / Rembrandt / Rubens	60

TABLE 3.3: Results of 3-class experiments.

Authors	Accuracy
Van Gogh / Monet / Toulouse Lautrec / Gauguin	75
Van Gogh / Monet / Toulouse Lautrec / Rembrandt	86.6667
Van Gogh / Monet / Toulouse Lautrec / Rubens	85
Van Gogh / Monet / Gauguin / Rembrandt	83.3333
Van Gogh / Monet / Gauguin / Rubens	73.3333
Van Gogh / Monet / Rembrandt / Rubens	78.3333
Van Gogh / Toulouse Lautrec / Gauguin / Rembrandt	68.3333
Van Gogh / Toulouse Lautrec / Gauguin / Rubens	61.6667
Van Gogh / Toulouse Lautrec / Rembrandt / Rubens	81.6667
Van Gogh / Gauguin / Rembrandt / Rubens	70
Monet / Toulouse Lautrec / Gauguin / Rembrandt	63.3333
Monet / Toulouse Lautrec / Gauguin / Rubens	56.6667
Monet / Toulouse Lautrec / Rembrandt / Rubens	75
Monet / Gauguin / Rembrandt / Rubens	63.3333
Toulouse Lautrec / Gauguin / Rembrandt / Rubens	51.6667

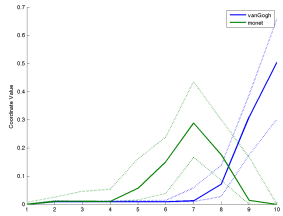
TABLE 3.4: Results of 4-class experiments.

Authors	Accuracy
Van Gogh/Monet/Toulouse Lautrec/Gauguin/Rembrandt	69.3333
Van Gogh/Monet/Toulouse Lautrec/Gauguin/Rubens	64
Van Gogh/Monet/Toulouse Lautrec/Rembrandt/Rubens	78.6667
Van Gogh/Monet/Gauguin/Rembrandt/Rubens	69.3333
Van Gogh/Toulouse Lautrec/Gauguin/Rembrandt/Rubens	61.3333
Monet/Toulouse Lautrec/Gauguin/Rembrandt/Rubens	56

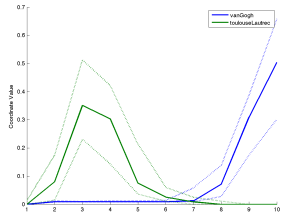
TABLE 3.5: Results of 5-class experiments.

Authors	Accuracy
Van Gogh/Monet/Toulouse Lautrec/Gauguin/Rembrandt/Rubens	62.2222

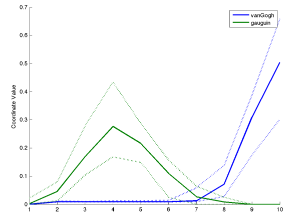
TABLE 3.6: Results of 6-class experiments.



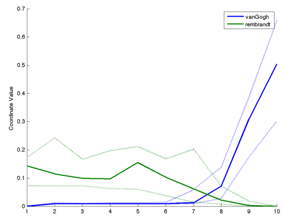
(A) Van Gogh / Monet



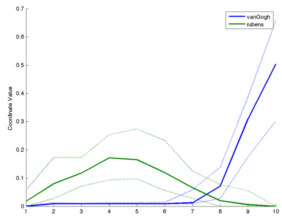
(B) Van Gogh / Toulouse Lautrec



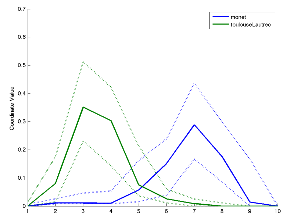
(C) Van Gogh / Gauguin



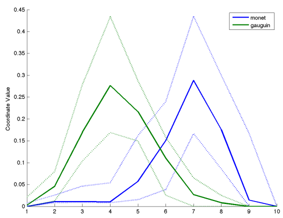
(D) Van Gogh / Rembrandt



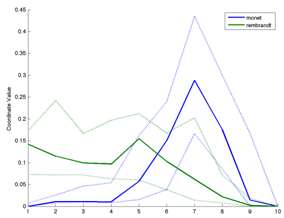
(E) Van Gogh / Rubens



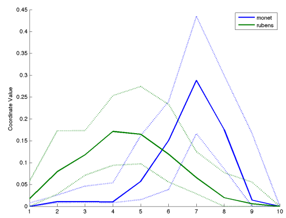
(F) Monet / Toulouse Lautrec



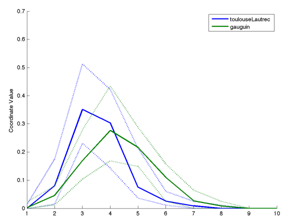
(G) Monet / Gauguin



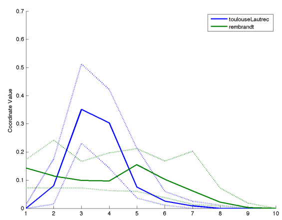
(H) Monet / Rembrandt



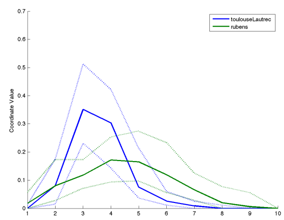
(I) Monet / Rubens



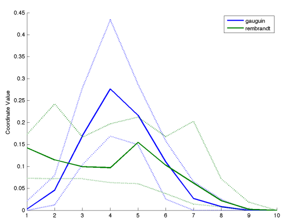
(J) Toulouse Lautrec / Gauguin



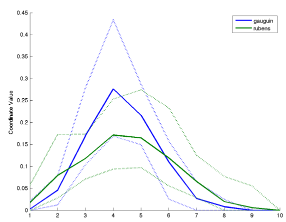
(K) Toulouse Lautrec / Rembrandt



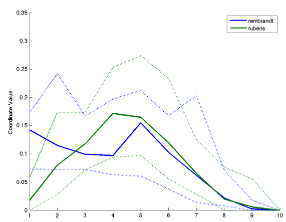
(L) Toulouse Lautrec / Rubens



(M) Gauguin / Rembrandt



(N) Gauguin / Rubens



(O) Rembrandt / Rubens

FIGURE 3.19: The data visualization of a 2-class experiment by a parallel coordinates plot

3.6 Discussion and Conclusion

3.6.1 Discussion

In this section we will discuss the advantages and disadvantages of our approach for painting classification. Several methods have been introduced in this thesis. Comparing with other methods introduced in section 2, the advantages of our approach are:

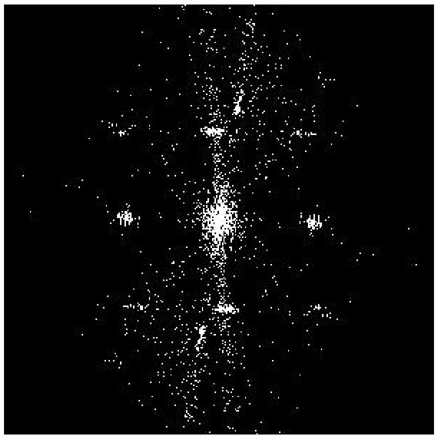
1. A high classification accuracy is achieved.
2. Paintings from different artists are analyzed and classified, which shows the method is generic.
3. The approach runs automatically after inputing a labelled training dataset.

However, the experiments also show some disadvantages which must be taken into account:

1. If the method is used to classify more artists, e.g., 10 artists, the accuracy will be lower.
2. The Fourier transform of brushstroke from different painters could be similar.
3. Other elements of the brushstroke texture will affect the Fourier transform, e.g., canvas pattern(Figure 3.20 shows the canvas pattern of the image will make additional high values in the frequency domain).

Then, the transformation from painting to digital image will also affect the results. Normally, the higher resolution the digital image we use, the better classification result should be obtained. But it is not the only standard, the hardware of painting digitalizing is also important. It is obvious that the digital image of the same painting will be different from a professional scanner or a mobile camera.

(A) canvas texture



(B) Fourier transform

FIGURE 3.20: Canvas texture and its Fourier transform

Based on current work, it is not likely that the mathematic methods mentioned in the thesis will replace art expert anytime soon, but they can assist in the work of analyzing paintings and improve the performance of human.

3.6.2 Conclusion

Image processing technology in painting analysis is becoming possible because high-resolution and richer data are becoming available. This thesis summarizes a treatment of the problem of painting classification, and the project was able to achieve the following tasks:

1. Fourier analysis was used for analyzing brushstroke texture and extracting important characteristics from it.
2. A simple and efficient method was formulated to create a feature vector of a painting.
3. A SVM model was used to classify paintings from several different artists.
4. A dataset was created that includes paintings from six different painters, and then, classification experiments were performed.

All of the experiments yielded encouraging but not perfect results. Using the visualization tools, the results clearly show the differences of the features from the painters.

To summarize, we used the discrete Fourier transform(DFT) for the brushstroke analysis to extract the properties. Section 1 gives an overview of the project and introduces the background and motivation. In section 2, we discussed other methods about the painting analysis. Section 3 introduced the basic knowledge used in the approach. In section 4, the details of the creation of the proposed approach were given. Finally, in section 5, we discussed the experiment datasets and gave the results of the experiments; then, we used visualization tools to analyze the results.

Chapter 4

Conclusion and Future work

In the preceding chapters, we studied the Fourier transform and digital image processing using FT, and described a 3D visualization tool for Fourier transform and some filters in the frequency domain. Then, we presented a method for classifying paintings by their creating artists using Fourier transform. The proposed experiments also provides similarity measures between the given painters.

The primary findings of our studies are:

1. 3D visualization could help to learn and understand Fourier transform of image.

2. 3D visualization could improve user interaction of filtering and noise removal of image in the frequency domain.

3. 3D visualization is limitedly useful for the analysis of complicated pattern.

4. Fourier transform is effective for brushstroke analysis.

5. The brushstroke size of the painting from one artist self is similar.

6. The brushstroke size could also be similar from different artists.

As demonstrated in chapter 3, most experiments obtained encouraging but not perfect results. And the classification method is still possible to be improved in many ways, better results could be achieved. With development, the 3D visualization is also possible to assist to create custom filtering to analyze complicated patterns of the image in the future.

4.1 Future Work

The approach of painting classification that we followed in this thesis could be improved in many ways:

1. A comparison between different parameters of the features extractions could improve the accuracy of the classification.
2. Using or combining different features (e.g., wavelet transform or color) in machine learning algorithms could enhance the results.
3. Using different machine learning algorithms such as Bag-of-Words(BoW) or K-Nearest-Neighbors(KNN) could enhance the results.
4. Using a larger training set or painting image of higher resolution could enhance the results.

Appendix A

Artworks

van Gogh:

Women on the Peat Moor — 1883

Portrait of a Prostitute — 1885

Garden with Courting Couples: Square Saint-Pierre — 1887

The Yellow House (The Street) — 1888

Orchards in Blossom, View of Arles — 1889

The Bedroom — 1888

Portrait of Etienne-Lucien Martin — 1887

Sunflowers Gone to Seed — 1887

Self-Portrait with Grey Felt Hat — 1887

Bank of the Seine — 1887

Snow-Covered Field with a Harrow (after Millet) — 1890

Almond Blossom — 1890

The Garden of Saint Paul's Hospital — 1889

View of the Alpilles — 1890

Undergrowth — 1889

Olive Trees — 1889

Daubigny's Garden — 1890

Tree Roots — 1890

The Vicarage at Nuenen — 1885

Night (after Millet) — 1889

HORSE CHESTNUT TREE IN BLOSSOM — 1887

Montmartre: behind the Moulin de la Galette — 1887

Exterior of a Restaurant in Asnieres — 1887

Wheat Field with a Lark — 1887

Piles of French novels — 1887

Self-Portrait, Winter — 1886

Landscape with Rabbits — 1889

Still Life with Lemons on a Plate — 1887

Quinces, lemons, pears and grapes — 1887

The Hill of Montmartre wi — th Stone Quarry — 1886

Monet:

Rouen Cathedral, West Facade, Sunlight — 1894

Woman with a Parasol - Madame Monet and Her Son — 1875

Jerusalem Artichoke Flowers — 1880

The Seine at Giverny — 1897

The Houses of Parliament, sunset — 1903

Palm Trees at Bordighera — 1884

Rapids on the Petite Creuse at Fresselines — 1889

Vetheuil in Summer — 1880

Haystacks (Effect of Snow and Sun) — 1891

Bouquet of Sunflowers — 1881

Dr. Leclenche — 1864

Regatta at Sainte-Adresse — 1867

The Parc Monceau — 1876

lle aux Fleurs near Vetheuil — 1880

Bridge over a Pond of Water Lilies — 1899

Water Lilies — 1916

Jean Monet on his Hobby Horse — 1872

The Strollers — 1865

The Cradle - Camille with the Artist’s Son Jean — 1867

The Bridge at Argenteuil — 1874

Ships Riding on the Seine at Rouen — 1872

Rouen Cathedral, West Facade — 1894

The Artist’s Garden in Argenteuil — 1873

Sainte-Addresse — 1867

Argenteuil — 1872

Interior, after Dinner — 1868

Waterloo Bridge, London, at Dusk — 1904

Bridge at Argenteuil on a Grey Day — 1876

Gauguin:

Ia Orana Maria (Hail Mary) — 1891

Still life with teapot and fruits — 1896

When Will You Marry — 1892

Two Women — 1901

Haere Mai — 1891

A Mythic Life in Painting — 1890

Sacred Spring: Sweet Dreams — 1894

The Month of Mary — 1899

Arlesiennes (Mistral) — 1888

Washerwomen in Arles — 1888

Mata Mua (In Olden Times) — 1892

Tahitian Landscape — 1897

Washerwomen — 1888

Two Tahitian Women — 1899

Landscape from Arles — 1888

Faturuma — 1891

Night Cafe in Arles — 1888

Landscape with peacocks — 1892

I Raro Te Oviri (Under the Pandanus) — 1891

The Willow Tree — 1889

Words of the Devil — 1892

Paysage — 1894

Eu haere ia oe — 1892

Three Tahitians — 1898

Women bathing (Dieppe) — 1885

The Bathers — 1897

Tahitian Women on the Beach — 1891

Still Life with Three Puppies — 1888

Still Life with Peaches — 1889

Spirit of the Dead Watching — 1892

Toulouse Lautrec:

Portrait of Octave Raquin — 1901

Moulin Rouge — 1891

Femme de maison close — 1894

L'Anglaise du Star du Havre — 1899

At the Moulin Rouge, The Dance — 1890

Madame Marthe X Bordeaux — 1900

Redhead (Bathing) — 1889

Artista com Luvas Verdes — 1899

At the Moulin Rouge — 1892

Woman seated in a Garden — 1891

In the Wings at the Circus — 1887

At the Circus: Entering the Ring — 1899

The Englishman at the Moulin Rouge — 1892

Abandonment (The pair) — 1895

The Dog — 1880

Marcelle Lender Dancing the Bolero in "Chilperic" — 1895

Monsieur Fourcade — 1889

An Englishman at the Moulin Rouge — 1892

The Ladies in the Dining Room — 1893

Equestrienne (at the cirque fernando) — 1888

Le Jeune Routy a Celeyran — 1882

Au Cafe: Le consommeteur et la cassiere chlorotique — 1898

In a private room at the Rat mort — 1899

Jane Avril dansant — 1893

Ala Toilette: Madame Poupoule — 1898

Maurice Joyant Somme bay — 1900

The Opera "Messelina" at Bordeaux — 1900

Self-portrait in front of a mirror — 1882

The Bed — 1893

The Streetwalker — 1890

Rembrandt:

Study of an Old Man with a Gold Chain — 1632

Self-Portrait — 1629

Portrait of Nicolaes Ruts — 1631

Saskia van Uylenburgh, the Wife of the Artist — 1638

Portrait of Herman Doomer — 1640

Man in Oriental Costume — 1632

Aristotle with a Bust of Homer — 1653

An Old Man in Military Costume — 1630

Self-portrait — 1658

Old Man with Beard — 1630

Portrait of a man, presumably Dr. Ephraim Bueno — 1647

Portrait of Johannes Wtenbogaert — 1633

Portrait of Marten Looten — 1632

Sophonisba receives the cup of poison — 1634

St. Bartholomew — 1661

Portrait of Margaretha de Geer — 1661

Portrait of a Man, probably a Member of the Van Beresteyn Family — 1632

Portrait of a Woman, probably a Member of the Van Beresteyn Family — 1632

Portrait of an old man — 1632

A Young Scholar and his Tutor — 1629

Old Man with a Gold Chain — 1631

Portrait of a Young Man — 1666

The Abduction of Europa — 1632

Portrait of Jacob de Gheyn (III) — 1632

Jeremiah Lamenting the Destruction of Jerusalem — 1630

Portrait of Baertje Martens — 1640

Self-Portrait with Two Circles — 1665

Flora — 1654

Rubens:

Marchesa Brigida Spinola Doria — 1606

The Assumption of the Virgin — 1620

The Fall of Phaeton — 1605

Agrippina and Germanicus — 1614

Decius Mus Addressing the Legions — 1616

The Meeting Between Abraham and Melchizedek — 1617

Daniel in the Lions' Den — 1615

Deborah Kip and her Children — 1629

Head of One of the Three Kings: Melchior, The Assyrian King — 1618

The Meeting of David and Abigail — 1630

Study of Two Heads — 1609

Rubens, His Wife Helena Fourment and One of Their Children — 1630

The Holy Family with Saints Francis and Anne — 1630

The Feast of Achelous — 1615

Venus and Adonis — 1630

A Forest at Dawn with a Deer Hunt — 1635

The Triumph of Henry IV — 1630

The Wolf and Fox Hunt — 1616

Sketch for the Coronation of the Virgin — 1640

The Wrath of Achilles — 1635

Adam and Eve — 1629

The Judgement of Solomon — 1617

Portrait of the Artist — 1623

The Garden of Love — 1633

The Judgement of Paris — 1639

The Union of England and Scotland (Charles I as the Prince of Wales) — 1633

Bacchanal auf Andros — 1635

Bibliography

Ahsan Ahmad Ursani Kidiyo Kpalma, Joseph Ronsin-Bhawani Shankar Chowdhry (2008). "Improved Texture Description with Features Based on Fourier Transform". In: URL: https://www.researchgate.net/publication/220731645_Improved_Texture_Description_with_Features_Based_on_Fourier_Transform.

Arnheim (1954). "Art and visual perception: A psychology of the creative eye". In:

ArtExperts (2017). "Painting Authentication". In: URL: https://www.artexpertswebsite.com/authentication.php.

Bower, Tim (2009). "Frequency Domain Filters". In: URL: http://faculty.salina.k-state.edu/tim/mVision/ch11-freq-domain/freq_filters.html.

"Fingerprint may lead to new da Vinci discovery" (1887). In: *The Associated Press*. URL: https://usatoday30.usatoday.com/news/world/2009-10-14-new-da-vinci-fingerprint_N.htm.

Gemignani, Zach (2010). "Better Know a Visualization: Parallel Coordinates". In: URL: http://www.juiceanalytics.com/writing/writing/parallel-coordinates.

Gogh, Vincent van (2009). "Exterior of a Restaurant in Asnieres". In: URL: https://www.vangoghmuseum.nl/en/collection/s0134V1962.

Hendriks, Ella and Shannon Hughes (2007). "Van Gogh's brushstrokes; marks of authenticity?" In: *Art, Conservation and Authenticities*. URL: https://

www.researchgate.net/publication/280559427_Van_Gogh's_brushstrokes_marks_of_authenticity.

ImageMagick (2014). "Fourier Transform Processing With ImageMagick". In: URL: http://www.fmwconcepts.com/imagemagick/fourier_transforms/fourier.html.

Johnson, C. Richard, Ella Hendriks, and Igor J. Berezhnoy (2008). "Image Processing for Artist Identification". In: *IEEE Signal Processing Magazine*. URL: http://ieeexplore.ieee.org/document/4545847/.

Li, Jia, Lei Yao, and Ella Hendriks (2011). "Rhythmic Brushstrokes Distinguish van Gogh from His Contemporaries: Findings via Automated Brushstroke Extraction". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence*. URL: http://ieeexplore.ieee.org/document/6042878/.

LI Wen-xin David Zhang, XU Zhuo-qun (2002). "Palmprint Recognition Based on Fourier Transform". In: URL: https://www.researchgate.net/publication/266456288_Palmprint_Recognition_Based_on_Fourier_Transform.

Lombardi, Thomas (2005). "The Classification of Style in Fine-Art Painting". In: *2005 Doctoral Dissertation*. URL: http://dl.acm.org/citation.cfm?id=1124009.

Matlab (2006). "parallelcoords". In: URL: https://de.mathworks.com/help/stats/parallelcoords.html.

OpenCV (2017a). "Histogram Equalization". In: URL: http://docs.opencv.org/trunk/d5/daf/tutorial_py_histogram_equalization.html.

— (2017b). "Introduction to Support Vector Machines". In: URL: http://docs.opencv.org/2.4/doc/tutorials/ml/introduction_to_svm/introduction_to_svm.html.

QtCompany (2013). "Qt Data Visualization Overview". In: URL: http://doc.qt.io/qt-5/qtdatavisualization-overview.html.

Rigau, Jaume et al. (2010). "Toward Auvers Period: Evolution of van Goghs Style". In: *Computational Aesthetics in Graphics, Visualization, and Imaging*. URL: http://dl.acm.org/citation.cfm?id=2381330.

Wikipedia (2017a). "Low-pass filter". In: URL: https://en.wikipedia.org/wiki/Low-pass_filter#Ideal_and_real_filters.

— (2017b). "Support vector machine". In: URL: https://en.wikipedia.org/wiki/Support_vector_machine.