Artistic style transformation of images

Kwan Ho Alan Cheng[1], Jiajun Li[2], Rixin Nie[3], Tam Van Phan[4], Praveen Vinny[5], Zhuo Wang[6], Rui Zhang[7]

{khac070[1], jl390[2], rn958[3], vtp990[4], pv562[5], zw750[6], rz687[7]}@uowmail.edu.au

School of Computer Science and Information Technology, University of Wollongong, Wollongong, New South Wales 2522, Australia

Abstract

The topic of this research is to implement the picture recognition and auto colouring which is based on the recognition algorithm and machine learning. Algorithms about machine learning and image recognition are trending in the field of artificial intelligence. Its popularity does not seem to be losing any momentum. For instance, PaintsChainer[6] website provides auto colouring service, gave a giant leap in the advances related to painting in 2016, and it has undoubtedly became a significant application in machine. Meanwhile, the cloud version API was opened by Google in February 2017. It is the service that help developers to integrate the function of image recognition. Analysing these trends, we intend to research and apply the image recognition and auto colouring in our project to apply artistic styling to transform images.

1. Project Background

Artificial intelligence (AI) is always a hotspot problem in the computer science since the terminology has been put forward in 1956 Dartmouth conference. Artificial intelligence is a technical science to research and develop theory, method, technique, and application system used to simulate and expand human intelligence. Artificial intelligence simulates the information process of human consciousness and human thinking. Artificial intelligence is not the human intelligence but it could be thinking like a human being or even surpass human's intelligence. Main research fields are artificial-intelligence robot, speech recognition, national language process, expert system, and image recognition.

Image recognition is one of the important branches of AI. It is based on the major characteristic of the image. Every image has its own characteristics. For example, capital letter A has a sharp and letter p has a circle. Human's eye always focus on the major characteristic of the image in the process of image recognition. Thus, it is important to get rid of redundancy information and focus on the key features. Computers could recognize the image by comparing the key features with the database like that of pattern match. However, not every image has their pattern in the database. Gestalt psychologist propose prototype matching model to deal this problem [1]. Prototype matching model focus on the similarity. The method abstract the similarity from the image. The similarity is called prototype which is used to recognize the image. Prototype matching model clarify how human analysis and process the similarity. Obviously, this model is better than pattern matching. However, this model is difficult to employ in computer programs. In modern industry, programs find useful information through image grey-scale difference.

Deep learning is another important branch of the AI. Deep learning is the fastest-growing field in artificial intelligence, helping computers make sense of infinite amounts of data in the form of images, sound, and text. Using multiple levels of neural networks, computers now have the capacity to see, learn, and react to complex situations as well or better than humans. Today's deep learning solutions rely

almost exclusively on NVIDIA GPU-accelerated computing to train and speed up challenging applications such as image, handwriting, and voice identification. There are numerous applications in deep learning. For instance, auto-colouring and modification of the style of the painting are recently out of scope of research. But they are obviously the useful applications of Deep learning.

Our proposed project will primarily have three parts. Initially, the application must recognize the character of the image provided by the user. And the application should complete the image. For example, if user draws a dragon horn and dragon wings, application should draw a dragon based on user's painting. It is worth mentioning that application should learn from user's painting style that the picture will be completed in the matching style. The second phase of the proposed project is to colour the whole picture. This process should be upon the basis of deep learning. The final phase is to modify the painting into different style like photorealism, conceptual art, impressionism and so on.

2. Project Plan

Even though, we will use Gantt chart to manage the project, the whole project can be divided into six phases which are elaborated as follows:

2.1 Feasibility Study and Investigation

The initial step is to do the research on how AI and deep learning works in images and to understand the current trends in this area. In this phase, we aim to enhance the background knowledge. We would be in need to spend an entire month to do the literature review. We may require further reading even after this time span to progress further in our project

2.2 Requirement Analysis

We will start with the requirement analysis post background study. During this phase, we will discuss about the plan of execution of the project. We would be in need to spend at least a week on this phase. But it may be further expanded based on the requirement. In a real-life environment, the requirements may change very frequently. Similarly, we may be in need to incorporate some features into the system to implement the trends in technology

2.3 System Design

We would be having a design in hand by the end of the requirement analysis phase. During the system design phase, we will make decisions about the environment of our project and the code structure of our system. We will split our system into several small subsystems making it easy to maintain. We will spend two to three weeks to do this.

2.4 Coding

We already have the structure defined for the subsystems before we start with this phase. We would be focussing on the implementation of the systems. We will jointly face and encounter the issues if any this phase. This phase may consume seven to eight weeks. While the developers are doing the coding, the testers would be defining the test cases. Moreover, the developers would be doing unit testing on each of the modules which are being developed.

2.5 Testing

In the phase of system design, we split our system into several subsystems to make our system easy to maintain. During the testing phase, we will execute the test cases designed by the testers. We will also perform regression testing, integration testing and user acceptance testing during this phase.

2.6 Project Release

The final project which underwent all phases of testing will be ready for demonstration to the public. The public can use the functionalities of the system at this phase. We can also accept the feedback from post production testing and can incorporate the changes in the system during this phase.

3. Risk Assessment

There can be various risks that may arise during the development of a system. We came across the possibilities of the risks as mentioned below while studying about the risks related to our project.

3.1 Software project risk

Software project developers are concerned about the factors that may lead to failure of software projects and how it's going to impact the project. The risk of a software project is that the uncertainties that may occur during the software development process can cause loss or impact on various other modules. If the project risk occurs during the project development period, it may affect project progress, increases project costs, and can even cause software projects to fail. Therefore, it is necessary to analyze the project risks and respond to risks to take appropriate measures to minimize the losses caused by the risks.

Software project risk mainly includes the following four categories:

1. Since the development team and customer communication is not clear, the customer demand is not clear. Such a risk may occur in the entire software development project life cycle. This primarily consists of change risks, process risks and installation and maintenance risks.
2. When the quality of management is not enough, or when the management lacks experience, poor communication and unreasonable distribution of tasks causes the risk. This includes the progress of risk, budget risk, management risk and information security risks.
3. Due to the lack of technical strength and development environment tools, there arises risks such as technical risks, quality risks, software design tool risks, software development tool risks and employee skill risks.
4. The internal and external changes in the environment causes the risk. This includes human resources risk, policy risk, market risk and marketing risk.

3.2 Risks in the project

There could be several risks arising during the design of a project. These risks can impact us during the development of the system.

3.2.1 Requirements change risk

The requirement change risk is that the demand has become the project benchmark, but the demand continues to change; the lack of definition of the requirements, so that further definition will expand the scope of the project; addition of additional requirements, ambiguous product definition; parts need more time than expected; Effective demand change management process, etc. The ways to prevent this risk is the need for team members with a high degree of cooperation and close collaboration, careful

allocation of team members' work when conducting requirements analysis and to effectively curb demand change. The software requirements change is the biggest villain in the software project development and implementation.

3.2.2 Technical risk

In the process of software project development and construction, strategic management technology is a very important factor. The project team must be in line with the actual requirements of the project, choose the right, mature technology and do not ignore the actual situation of the project. Significant technical risks include the existence of software architecture system problems resulting in the completion of the software product to fail to achieve the project target and risks due to the project implementation with a new technology with defects in the technology or due to the lack of knowledge in the technology, resulting in the development of the product with poor performance and poor quality.

The way to prevent this risk is to use the technology necessary for the project, in the technical application before the relevant personnel to carry out good technical training. First of all, do a good job in the technical review of the various stages, through the collective wisdom to ensure that the feasibility of the technology used in the project and the correctness of the technical program. Second, the use of new technologies to be careful, to step by step, as far as possible the use of mature technical solutions to complete the software development work. Third, in the technical innovation and technical risk between the balance, and do a good job of innovative technology research and testing. The need to evaluate the various technologies used in the software project process, software project management in the development of software development plans must consider these factors, and make a reasonable trade-off decision.

3.2.3 Tool risk

Software project development and implementation process, the necessary management tools, development tools, test tools, etc. can be timely in place. The project team should consider the risk factors to see if the tool version is consistent with the project requirements. Some software projects are multi-user concurrent application systems, the system requirements for high performance, the project team will need to pay attention to the project's performance risk.

The way to prevent this risk is usually to implement the tools at the start-up phase of the tools or possible alternative tools, before the need for the use of these tools (usually need a month or so in advance) to track and implement the tools in place. Before the project development and design and build the system infrastructure and performance testing to ensure that the framework to meet the performance indicators and then follow-up work. And the team's technology is biased towards that tool.

4. Project Plan Timeline

Team Formation	Week 1 – Week 3
Framing Code of Conduct	Week 4
Background Study and Project Proposal	Week 5 – Week 6
Creation of Software Requirements Specification	Week 7 – Week 10
Analysis and design of system blue print	Week 11 – Week 13
Design of Test Cases	Week 14 – Week 15
Project Progress Analysis and Submission	Week 16
Integration Testing and Project Documentation	Week 17 – Week 20
User Acceptance Testing for the Project	Week 21 – Week 22
Project Thesis Preparation and Demonstration	Week 22 – Week 26

5. Teamwork: Roles and Responsibilities

Team Member	Role
Rixin Nie	Manager and Developer
Tam Van Phan	Developer
Praveen Vinny	Project Documentation and Meeting Minutes keeper.
Jiajun Li	Documentation
Kwan Ho Alan Cheng	Tester
Zhuo Wang	Tester

1. Rixin will be the manager. Rixin will also act as the first escalation point for any issues within the project.
2. Rixin will also do the development.
3. Tam will also do coding with Rixin. While Rixin is concentrating on the Web and UI design, Tam will be concentrating on the backend coding.
4. Praveen will be responsible for keeping the minutes for each of the meetings.
5. Praveen will also do the entire documentation for the project along with Jiajun Li.
6. Even though Rixin and Tam will do unit testing on the modules they are developing, Alan and Zhuo Wang will write test cases and thoroughly examine each module developed in every phase to validate and verify the requirements.
7. The members would be in need to take additional responsibilities in times of urgent need.
8. During the final phases of the project, if there arises a need, Praveen will also do coding with Rixin and Tam.
9. Jiajun will help Alan and Zhuo Wang in test case execution during the integration of various sub modules in the project.

6. Conclusion

In the real-world, painters doing fine art, especially painting, have mastered the skill to create unique visual experiences through composing a complex interplay between the content and style of an image. So, the algorithmic basis of this process is yet unknown and there exists no artificial system with similar capabilities. However, in other key areas of visual perception such as object and face recognition near-human performance was recently demonstrated by a class of biologically inspired vision models called Deep Neural Networks [2]. In this project, we introduce an artificial system based on a Deep Neural Network that creates artistic images of high perceptual quality.

7. References

1. Leon A. Gatys, Alexander S. Ecker, Matthias Bethge: "A Neural Algorithm of Artistic Style." CoRR abs/1508.06576 (2015)
2. Leon A. Gatys, Alexander S. Ecker, Matthias Bethge: "Image Style Transfer Using Convolutional Neural Networks". CVPR 2016: 2414-2423
3. The article "Convolutional neural networks for artistic style transfer" extracted from https://harishnarayanan.org/writing/artistic-style-transfer/ on April 4, 2017.
4. The article "TensorFlow Implementation of "A Neural Algorithm of Artistic Style" extracted from http://www.chioka.in/tensorflow-implementation-neural-algorithm-of-artistic-style on April 5, 2017.
5. The article "Deep neural networks can now transfer the style of one photo onto another" extracted from http://www.theverge.com/2017/3/30/15124466/ai-photo-style-transfer-deep-neural-nets-adobe on 5th April 5, 2017.
6. The tool 'PaintsChainer' available in https://paintschainer.preferred.tech/.