

**MINISTRY OF EDUCATION AND TRAINING
HO CHI MINH UNIVERSITY OF TECHNOLOGY AND EDUCATION
FACULTY FOR HIGH-QUALITY TRAINING**



HCMUTE

PROJECT IN SOFTWARE ENGINEERING

**CLASSIFYING THE DESIGN STYLE
OF AN INTERIOR OBJECT USING
DEEP LEARNING**

Advisor: Dr. Nguyễn Thiên Bảo

Students:

Nguyễn Đan Trường	19110064
Đặng Ngọc Trường Giang	19110104
Khúc Nguyễn Huy Cường	19110107

HO CHI MINH CITY - 2023



CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM

Độc lập – Tự do – Hạnh phúc

.....***.....

Ho Chi Minh, Day...Month...Year 2023

SPECIALIZED ESSAY MISSION

Full name: Nguyễn Đan Trường

Student ID: 19110064

Class: 19110CLA1

Full name: Đặng Ngọc Trường Giang

Student ID: 19110104

Class: 19110CLA1

Full name: Khúc Nguyễn Huy Cường

Student ID: 19110107

Class: 19110CLA1

Major: Software Engineering

Instructor guide: Dr. Nguyễn Thiên Bảo

Date of project receipt: 1/10/2022

Project submission date: 2/01/2023

- 1. Project Name:** Classifying the design style of an interior object using deep learning.
- 2. Data, research paper:** Datasets from instructor guide, research papers in Deep Learning, CNN models from instructor and self-search.
- 3. Project contents:**
 - Building prediction model based on Interior static images using ResNet50, VGG19, InceptionV3, Xception.
 - Building an intuitive website for Interior image classification
- 4. Products:** A website for design style of an interior object classification

CHAIR OF THE PROGRAM

INSTRUCTOR

Ho Chi Minh, Day...Month...Year 2023

INSTRUCTOR COMMENTARY

Full name: Nguyễn Đan Trường Student ID: 19110064 Class: 19110CLA1

Full name: Đặng Ngọc Trường Giang Student ID: 19110104 Class: 19110CLA1

Full name: Khúc Nguyễn Huy Cường Student ID: 19110107 Class: 19110CLA1

Major: Software Engineering

Project name: Classifying the design style of an interior object using deep learning

Reviewer teacher: Dr. Nguyễn Thiên Bảo

COMMENTARY

1. Topic content and workload

- Theory
 - Study and research about design style of an interior object
 - Research about machine learning and deep learning: ResNet50, VGG19, InceptionV3, Xception.
 - Propose a method of applying deep learning to solve the problem of design style of interior object classification
- Experiments
 - Using YOLOV5 for object segmentation to collect a dataset of interior objects labeling on design styles.
 - Implement 04 models for classifying types of design style of interior object: ResNet50, VGG19, InceptionV3, Xception with the accuracy of 90%.
 - Build an application using the above model on a simple website.
- Links
 - <https://github.com/huycuong291/Classifying-the-design-style-of-an-interior-object-using-deep-learning>
 - <https://detect-style-interior.vercel.app/>

2. Advantages

- The interior object dataset can be reused.
- The models have been trained with high accuracy.
- The website is friendly and easy to use.
- Students have the capability of self-studying and active learning.

- The project can be accomplished as scheduled.
3. Disadvantages
- Limited time for developing project.
 - The models have not been modified.
4. Recommend for protection or not?
-
5. Type rating:
-
6. Grade:(Text:)

Ho Chi Minh, Jan 2nd, 2023

Instructor's commentary

Ho Chi Minh, Day...Month...Year 2023

REVIEWER'S COMMENTARY

Full name: Nguyễn Đan Trường Student ID: 19110064 Class: 19110CLA1

Full name: Đặng Ngọc Trường Giang Student ID: 19110104 Class: 19110CLA1

Full name: Khúc Nguyễn Huy Cường Student ID: 19110107 Class: 19110CLA1

Major: Software Engineering

Project name: Classifying the design style of an interior object using deep learning

Reviewer teacher: Mr. Hoàng Văn Dũng

COMMENTARY

1. Regarding the content of the topic and the volume of implementation:

.....
.....

2. Strength:

.....
.....

3. Drawback:

.....

4. Recommend for defending or not?

.....
.....

5. Grade:

6. Mark:

Ho Chi Minh, Jan 2nd, 2023
Reviewer's commentary

ACKNOWLEDGEMENT

We would like to offer our heartfelt appreciation to the Department of High-Quality Training, Ho Chi Minh City University of Technology and Education, and all the professors and teachers who have actively taught and aided me during my study and research.

In particular, we would like to express our heartfelt thanks to Mr. Nguyen Thien Bao for personally instructing, leading and establishing all suitable conditions to let me carry out the thesis. He helped us a lot in picking a topic and directing us to understand, investigate the theory and practice the issue. During the thesis implementation, the knowledge that the teacher provided us was extremely valuable, not only assisting us in completing the thesis but also supplementing us with a significant amount of core knowledge.

However, due to our limited professional expertise and limitations in our personal experience, the substance of the report is unavoidable. We would like to get your thoughts and more direction. She guides and criticizes to make this report more thorough.

Ho Chi Minh city, December 09th, 2022

Đặng Ngọc Trường Giang

Nguyễn Đan Trường

Khúc Nguyễn Huy Cường

PREFACE

Computer technologies are becoming increasingly popular as the information technology era progresses, and they play an important role in human life. Machines are gradually replacing humans in heavy manual work.

Besides, the field of artificial intelligence arises as an unavoidable evolution. Computers can intelligently learn, interpret, and process at the level of human thought thanks to artificial intelligence (AI).

Deep Learning - Machine Learning is the most rapidly growing AI, and it is inspired by the human brain - the connections between neurons. Deep Learning has been applied in many real-life problems, breaking the limitations of computers, helping machines to handle or almost exactly like humans.

Applying that practice, our team would like to present a report on the application topic of Deep Learning which is "Classification of interior design styles applying deep learning".

LIST OF TABLES

Table 1: Data statistic.....	41
Table 2: Pre-built models provided by Keras	46

LIST OF FIGURES

Figure 1: Classification problem.....	16
Figure 2:Relationship between AI, ML and DL	26
Figure 3: Convolution equation	27
Figure 4: Examble of Kernel.....	28
Figure 5: Convolutional Neural Network	29
Figure 6: The scope of work automatically exploits the devices in interior design and their design features.	32
Figure 7: A high-level overview of the Early-fusion Blending architecture	33
Figure 8: Architecture of the team's DeepStyle-Siamese network	34
Figure 9: An overview of the proposed method.	35
Figure 10: VGG19 architecture	36
Figure 11: Exception and Inception V3 architectures	37
Figure 12: Resnet50 architecture	38
Figure 13: PATH of application	40
Figure 14: Load data from directory	40
Figure 15: Python feature.....	42
Figure 16: Google colab.....	42
Figure 17: Jupyter	43
Figure 18: Flask	44
Figure 19: ReactJS	44
Figure 20: YOLOv5	45
Figure 21: Xception Model	48
Figure 22: Code for train model.....	48
Figure 23: Code for plot accuracy	49
Figure 24: Code for plot loss	49
Figure 25: Loss and accuracy graph	49
Figure 26: Confusion matrix	50
Figure 27: System proceed overview	51
Figure 28: Welcome screen of the application	52
Figure 29: Main screen of the application	53
Figure 30:Test API with Postman.....	53
Figure 31: Main screen created by ReactJS	54
Figure 32: Feature introduction screen	55
Figure 33: Group member introduction screen	55
Figure 34: Demonstration for application.....	56
Figure 35: Evaluate crop image	57
Figure 36: Client evaluate wrong style 1	57
Figure 37: Client evaluate wrong style 2	58

Figure 38: Client evaluate right style..... 58

TABLE OF CONTENTS

CHAPTER 1: DESIGN STYLE CLASSIFICATION PROBLEM	15
1.1 House design process	15
1.1.1 Research reality.....	15
1.1.2 Problem	15
1.2 Solution	15
1.3 Objects and scope.....	15
1.3.1 Objects	15
1.3.2 Popular design styles	16
1.3.2.1 ArtDecor	16
1.3.2.1.1 What is ArtDecor?.....	16
1.3.2.1.2 Characteristics	16
1.3.2.1.3 Features	17
1.3.2.2 HiTech.....	17
1.3.2.2.1 What is HiTech?.....	17
1.3.2.2.2 Characteristics	17
1.3.2.2.3 Features	18
1.3.2.3 Indochina.....	19
1.3.2.3.1 What is Indochina?.....	19
1.3.2.3.2 Characteristics	19
1.3.2.3.3 Features	19
1.3.2.4 Industrial	20
1.3.2.4.1 What is industrial?.....	20
1.3.2.4.2 Characteristics	20
1.3.2.4.3 Features	21
1.3.2.5 Scandinavian	22
1.3.2.5.1 What is Scandinavian?	22
1.3.2.5.2 Characteristics	22
1.3.2.5.3 Features	22
CHAPTER 2. OVERVIEW ABOUT MACHINE LEARNING, DEEP LEARNING	24

2.1 What is Machine Learning?	24
2.1.1 Basic problems in Machine Learning	24
2.1.1.1 Classification.....	24
2.1.1.2 Regression.....	24
2.1.1.3 Machine translation.....	24
2.1.1.4 Clustering	24
2.1.1.5 Data completion	24
2.1.2 Classify Machine Learning Algorithms.....	25
2.1.2.1 Supervised learning.....	25
2.1.2.2 Unsupervised learning	25
2.1.2.3 Semi-supervised learning.....	25
2.1.2.4 Reinforcement learning.....	25
2.2 What is Deep Learning?.....	25
2.3 Convolutional Neural Network	26
2.3.1 Convolution integral	26
2.3.2 Convolutional Neural Network.....	28
2.3.3 Basic classes in CNN	29
2.3.3.1 Convolutional layer.....	29
2.3.3.2 Pooling layer	29
2.3.3.3 Relu layer	29
2.3.3.4 Fully connected layer.....	30
2.3.4 Structure of CNN	30
CHAPTER 3: DEEP LEARNING APPLICATION TO SOLVE THE PROBLEM OF CLASSIFICATION OF INTERIOR DESIGN STYLE.....	31
3.1 Related works.....	31
3.1.1 Research on “Accessing and extracting the characteristics of the interior”	31
3.1.1.1 Research Summary	31
3.1.1.2 Results of the research	31
3.1.2 Research on “Multimodal search engine for fashion and interior design”	32
3.1.2.1 Research Summary	32
3.1.2.2 Results of the research	32

3.2 Suggested method	35
3.3 Models	36
3.3.1 VGG19	36
3.3.2 InceptionV3	36
3.3.3 Xception	38
3.3.4 Resnet50	38
CHAPTER 4: EXPERIMENTS AND RESULTS	39
4.1 Dataset	39
4.1.1 Reference source for data	39
4.1.2 Data collection	39
4.1.2.1 Why is data collection important?	39
4.1.2.2 Data collection operations	39
4.1.2.3 Prepare, Explore, Clean, and Consistency of Data	39
4.1.3 Description of the dataset	41
4.2 Environment and Tools	42
4.2.1 Python	42
4.2.2 Google Colab	42
4.2.3 Jupyter	43
4.2.4 Flask	44
4.2.5 ReactJS	44
4.2.6 YOLOv5	45
4.3 Implementation process	45
4.3.1 Choose the train model	45
4.3.2 Train the selected model with the dataset	47
4.3.2.1 Keras's pre-built models	47
4.3.2.2 Configure for the model	47
4.3.2.3 Train the model	48
4.3.2.4 Evaluate the model	48
4.3.3 Web application development	51
3.3.1 Create APIs with high-accuracy models	51

3.3.2 Create application interface using ReactJS.....	54
4.4 Experimental results.....	58
CHAPTER 5: CONCLUSIONS AND FUTURE WORKS	59
5.1 Summary	59
5.2 Limitations	59
5.3 Future works.....	59
REFERENCES	60

CHAPTER 1: DESIGN STYLE CLASSIFICATION PROBLEM

1.1 House design process

1.1.1 Research reality

People's quality of life is increasingly improving as technology advances. Aside from such advancement, people's quality of life has significantly increased. The demand for housing has always been an essential necessity, and it is currently expanding. A house must not only be a location full of facilities to live in, but it must also be beautiful, convenient to use, safe, inexpensive, and affordable. Quality, comprehensive, and intricate design is never easy. As a result, we investigated the present house design process.

The modern house design process typically consists of five essential steps:

- Customers get advice, make requests, and share ideas with brokers.
- Brokers establish touch with designers and get consumer needs.
- The designer receives the request, designs it, and delivers it to the manufacturer.
- Manufacturers receive designs, assess feasibility, compile data, and provide estimates to clients.
- Customers obtain manufacturer information, request adjustments, and approve.

The process is iterated until the consumer has reached an agreement with the brokers, designers, and manufacturers.

1.1.2 Problem

The aforementioned method experienced some significant issues as a result of the survey:

- Idea disagreement.
- I'm wasting a lot of time.
- A huge waste of money.
- In the exchange process, there is a lack of information.

1.2 Solution

The challenge of using information technology to the problems of house design is to:

- Time has been reduced.
- Reduce the expenses.
- Consistency of information

Need a solution to help with design and identifying client wants.

1.3 Objects and scope

1.3.1 Objects

The topic focuses on the specific research in the realm of information technology, namely deep learning in analysis to assist customer requirements identification.

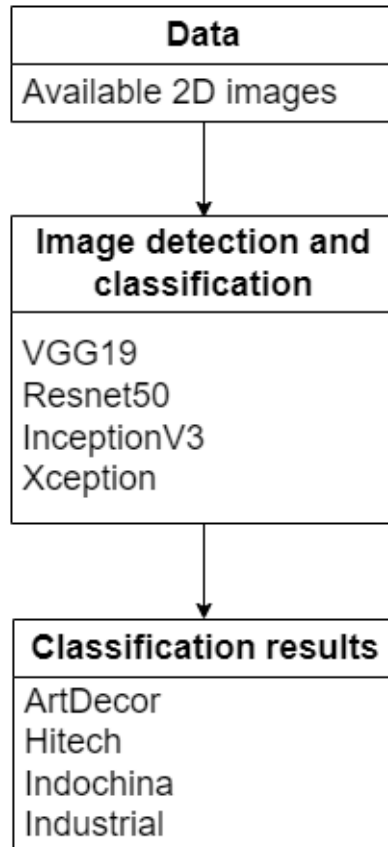


Figure 1: Classification problem

1.3.2 Popular design styles

1.3.2.1 ArtDecor

1.3.2.1.1 What is ArtDecor?

The ArtDecor style is an art and decoration school that originated in Paris in the 1920s and extended around the world in the 1930s.

ArtDecor strives for basic lines and classic geometric blocks in space, influenced by Cubism, to create a powerful and unique design, similar to German architecture today. before.

1.3.2.1.2 Characteristics

This style's basic background would include:

- Flowers and leaves with animal skins.
- Horizontal lines are preferred.
- Parallelograms, rectangles, and other angle quadrilaterals
- Image of a stylized animal model.
- Sharp geometric angles or representations of stylized architecture
- Military and zigzag patterns
- Some are distinguished by achievements in trade, science, and technology.

1.3.2.1.3 Features

About color:

- ArtDecor colors are often vivid and high contrast, such as yellow (bright gold or vintage), red, green, blue, rose silver, black, and other iridescent hues.
- Furthermore, colors that contrast with the sheen of the wood and lacquered interiors are popular in this design.

Concerning the material used:

Some materials, such as stainless steel, glass, or animal skin, will be unavoidable in this ArtDecor style. To enhance the luxurious and dazzling beauty, extremely expensive materials such as marble, rare wood, and so on are employed.

Concerning the furniture:

- Not crowded and intricate, but simple, kind, forceful, and charming.
- The illustration is simple.

Details such as materials, lighting, and accessories:

- Rugs with huge cubes were frequently used to cover hardwood floors.
- Crystals are frequently used in decorative lights to keep them looking bright and fresh.
- Ornamental textiles provide decorative value to an area by using contrasting forms and colors.

1.3.2.2 HiTech

1.3.2.2.1 What is HiTech?

Hi-tech architecture grew rapidly in the late twentieth and early twenty-first centuries. This movement expanded fast throughout numerous countries. This style thrived in both Japan and the West.

Hitech is also known as High Technology. This design constantly employs modern materials, equipment, furniture, and goods that incorporate cutting-edge technology into space architecture.

1.3.2.2.2 Characteristics

Hitech, like many other interior designs, has essential qualities that set it apart from the others:

- Display aesthetic value by utilizing cutting-edge technologies.
- The area is contemporary because of the use of simple colors.
- Simple interior design communicates freedom and refinement.
- Ensure optimal comfort.

1.3.2.2.3 Features

Colors that are simple:

Hitech interior design always combines incredibly basic colors with a lot of lines. Rustic hues are constantly fashionable and used more. Gray, white, and black are the most common hues used in design. This Neutral hue gives a distinctive living environment, boosts color contrast, and attractiveness when utilizing high-quality materials.

Minimalist design:

Do not limit furniture and colors as much as the minimalist interior design style does; instead, emphasis on simplicity; do not need fussy and intricate as the Classic style does. Hitech interior designers radiate simplicity, liberality, and refinement at all times.

Simultaneous minimalism is conveyed via colors, layouts, and ornamental themes; it is this oneness that gives the room life. The interior designs are straightforward but vibrant.

Materials used:

Materials such as glass, metal, stone, and others are commonly employed in interior design and construction when artificial materials and high technology are used.

Because these material lines are uniformly flat and straight, they convey power and decisiveness. Hitech-style furniture's construction always emits ethereal beauty from the inside out, from the exterior to the inside.

Furniture:

Hitech furniture is not fussy in terms of lines and details, and it is not overly large. This style's interior is fairly basic, with exquisite lines and no bending or clumsiness. The simplicity of this design adds independence and refinement to each homeowner's aesthetic.

Furniture is characterized by straight lines, smooth surfaces, and a limited number of ornamental elements. Concentrate on harmony rather than bright colors or intricate motifs. Many people who want simplicity and liberality in their homes are opting for this interior style.

Lighting for relaxation:

Using technological aspects to generate a spectacular light effect that clearly demonstrates the power and gentleness emanating from the inside out. Architects frequently employ recessed lights and downlights in conjunction with the interior to create a dazzling impression of a light-filled room.

This light has the role of aiding in the organization of space, boosting the aesthetic impact, and ensuring that the shared living area is attractive, airy, and safe.

Smart electronics:

Using smart technology in interior design appears to be a distinguishing aspect of the Hitech interior design style. To develop a particular style, the architect uses a harmonic blend of architectural and technical technologies. The furniture that is used has a certain purpose.

In space, smart equipment such as automated blinds, lighting, smart TVs, and automatic remote-control systems are commonly employed. These objects, when arranged in a scientific and logical manner, guarantee a contemporary and comfortable living for the owner.

1.3.2.3 Indochina

1.3.2.3.1 What is Indochina?

Indochine style is a beautiful blend of Eastern and Western cultures, specifically the two major civilizations of mankind: China and India.

Indochinese interior design chooses features that convey Vietnam's national character. Simple, elegant, but providing complete amenities in contemporary life to provide consumers comfort and convenience

1.3.2.3.2 Characteristics

When beds, countertops, and other furniture replace tables and chairs, the rustic, rustic qualities are combined with the most basic technology.

The blend of contemporary beauty and modern European current French interior style is tropicalized by indigenous identity and climate appropriate.

China's and India's styles are contrasting, creating attraction while also complementing each other's beauty.

1.3.2.3.3 Features

Colors used in Indochina interior design:

The Indochina design style uses neutral hues throughout the interior, such as light yellow, cream yellow, and white, to create a pleasant atmosphere appropriate for Vietnam's tropical environment.

Furthermore, other locations utilize warm, humid tropical hues to make a significant impact, such as yellow-orange, purple, red, and so on.

Material used:

Indochinese interior design incorporates wooden, bamboo, brick, and other bright Asian elements, bringing everyone closer and more intimate.

Patterns used:

Patterns are now common emblems of Indochinese style in interior design, and they generate extremely distinct traits such as:

- Ky Ha motifs
- Rectangle pattern
- Still life textures
- Textures of flowers, leaves, and fruit
- Animal motifs

Traditional Vietnamese round reliefs and statues:

Included symbols such as:

- Buddha statue: a religious symbol, a symbol of purity and peace.
- Breeds, puppets: these are folk symbols.
- Four spirits: simulate Long, Lan, Quy, and Phung animals that bring a lot of luck.
- Lotus: from the time period, the symbol of purity and purity of Buddhism.
- Chrysanthemum: idyllic, high bar, discreet and durable.
- Bodhi tree: Bodhi tree symbolizes the great enlightenment of Buddha.

Furnitures:

There are items in Indochina's interior design such as mahogany, counters, and screens that represent the imprint of local characteristics and culture on the French way of life.

1.3.2.4 Industrial

1.3.2.4.1 What is industrial?

Born in the early 20th century during the European Revolution Recession. The idea is to transform these structures into neighborhoods to accommodate people's housing demands. Back to the basics, simplicity, and primitiveness are all hallmarks of industrial design.

1.3.2.4.2 Characteristics

- Neutral color palette: Industrial spaces don't simply use shades of white like other minimalist designs. It utilizes the spectrum of whites, greys, blacks, as well as neutral tones of brown.
- Embrace natural light: Industrial decor typically features large natural windows with black panes, sometimes in a grid pattern.
- Highlight architectural materials: The industrial style typically has open floor plans and high ceilings. Instead of using drywall or wallpaper, buildings feature exposed brick, concrete floors, industrial pipes, and visible ductwork.
- Repurposed materials: A wooden coffee table with castor wheels, bookshelf made of reclaimed materials, and repurposed dining tables are a staple of industrial style and a great way to bring natural elements into the metal-heavy materials.
- Bare light bulbs: Edison bulbs hanging solo or in a decorative group as a chandelier are common elements of industrial home decor. If you prefer something less bare, large metal domes, a pendant light, or floor lamps that leave the bulb visible are also popular industrial design ideas.

- **Graphic lines:** Whether it's a windowpane or an iron wheel on bar stools, this style loves clean, graphic lines, particularly created with black metals, rather than sinuous lines and patterns. This is not to be confused with the graphic and highly stylized look of art deco, which highlights bold colors and metallics.
- **Create coziness with natural textiles:** Fabrics like weathered leather or linen work well in these environments for both furniture and accents, creating a lived-in and cozy feel.

1.3.2.4.3 Features

Home wall decoration design: The Industrial style is characterized by raw walls, natural wood-clad walls, polished concrete walls, etc., which creates a simple yet opulent and appealing imitation of a factory.

These walls make people feel a sense of familiarity, closeness, a bit of classicism and have very high artistic value.

Layout of windows and lighting:

Light is a very important factor, the main color used for space and furniture of this style is deep and dark tones so light is extremely important. If there is a lack of light the room will become very dark.

The use of light bulbs is a recognizable feature of this style. The design of large windows to take advantage of natural and healthy light sources is also one of the highlights.

Simple and minimal space layout:

Industrial interior design style is unique, the design seems to be simple but full of meaning. This style requires a large area of the floor and less furniture for the room. Since the industrial style colors are dark and the furniture and pieces are very strong, you only need to highlight some of the decor of the apartment.

Use Industrial style furniture:

Furniture is often used in dark tones. Metallic materials are also painted black to create a strong look. In addition, in industrial design, leather upholstery is often used such as stools, or sofas, etc.

The interior of this style is required to be as minimal as possible, trees can also decorate a little to optimize the living space in the best way.

Color in design:

Some of the popular colors in industrial design include white, gray, navy, black and dark brown wood.

You can also change the color shades to warm up a bit, but don't overdo it if you don't want to lose the inherent beauty of this style.

Highlight:

The highlight is created from the contrast and creativity, you can flexibly choose the decoration items.

1.3.2.5 Scandinavian**1.3.2.5.1 What is Scandinavian?**

Scandinavian design, which originated in Northern Europe, combines three factors in a harmonious way: beauty, minimalism, and practicality. Due to its simplicity, warmth, and ventilation, which create a comfortable living area, this style is highly well-liked.

As time goes on, architects from all over the world find limitless inspiration in Scandinavian interior design.

1.3.2.5.2 Characteristics

- Natural wood, leather, and fur are used to make the materials.
- Use white and natural hues.
- furniture and décor that are minimalist.

1.3.2.5.3 Features**Material used:**

A consistent element in Nordic home design is wood. Wooden decorations give a home a lively, inviting feel.

Stone is another significant component of Scandinavian design, in addition to wood. Pure white stone is typically utilized for Scandinavian home design in prominent locations like wall cladding and kitchen surfaces.

The majority of the materials used in Nordic interior design emphasize simplicity. Fur adds a luxurious touch to the room, and is frequently utilized as carpeting or wall art.

Color used:

In Scandinavian design, color is the focal point, with white as the dominant hue. The white gamut makes people feel as though their height has expanded since it resembles the color of snow

In order to break up the monotony of the room and give it a more youthful feel, this style can also combine white with other color schemes like cream, black, light gray, turquoise, etc. When attempting to design a space in the lovely Nordic style, lightness and visual neutrality are the requisites.

The light:

For Scandinavian, to get natural light sources, most Nordic style designs combine a wide window system with thin white curtains.

For Scandinavian, to get natural light sources, most Nordic style designs combine a wide window system with thin white curtains.

The use of decorative lighting like rattan lamps, copper lamps, or paper lanterns ensures the innate lightness and comfort of Nordic style while also maximizing the use of artificial light sources.

Decorational components:

The use of natural materials, usually bamboo and rattan, is becoming more and more common in interior designs in the Nordic style. In addition, the presence of trees and feng shui plants helps the space feel less overwhelming.

For the Scandinavian style, fireplaces are a must-have item because they not only bring warmth but also closely link with the floor and ceiling, making your living space complete and perfect.

With a simple criterion, the motifs used to decorate the Scandinavian design style also become simple, often with plaid and checkered motifs. However, in order to create depth for interior spaces, it is now time to start using basic white and black tones or geometric motifs inspired by peoples around the world.

Furniture:

Each interior space's furniture arrangement needs to be done in the proper proportions for the corresponding design theme.

Wooden folding chairs or big sofa sets can be used as accents in Scandinavian design.

CHAPTER 2. OVERVIEW ABOUT MACHINE LEARNING, DEEP LEARNING

2.1 What is Machine Learning?

Machine Learning is the subfield of computer science, that “gives computers the ability to learn without being explicitly programmed” – Wikipedia

2.1.1 Basic problems in Machine Learning

Many complex problems can be solved with machine learning. Here are some common problems:

2.1.1.1 Classification

Classification is one of the most studied problems in machine learning. In this problem, the program is asked to determine the class/label of a data point among different labels.

Common classification problems: handwriting image classification, spam email classification...

2.1.1.2 Regression

A problem is said to be regression if the label is not divided into groups but is a specific real value (the range of values is continuous).

Most forecasting problems (stock prices, house prices, ...) are often classified as regression problems.

2.1.1.3 Machine translation

In a machine translation problem, a computer program is asked to translate a passage in one language into another. The training data are bilingual text pairs. These documents may include only the two languages under consideration or additional intermediate languages. The solution to this problem has recently made great strides based on deep learning algorithms.

2.1.1.4 Clustering

Clustering is the problem of dividing data X into small clusters based on the relationship between the data in each cluster. In this problem, the training data has no labels, the model automatically divides the data into different clusters.

This is similar to asking a child to cluster puzzle pieces of different shapes and colors. Even though children don't know which pieces correspond to which shape or color, they will most likely still be able to sort the puzzle pieces by color or shape.

2.1.1.5 Data completion

A dataset can have many features, but feature collection for each data point is sometimes not feasible. For example, a photo may be scratched causing many pixels to be lost or information about the age of some customers may not be collected. Data completion is the problem of

predicting those missing data fields. The task of this problem is based on the correlation between data points to predict the missing values. Recommendation systems are a good example of this type of problem.

2.1.2 Classify Machine Learning Algorithms

2.1.2.1 Supervised learning

A machine learning algorithm is called supervised learning if the building of a model that predicts the relationship between input and output is performed based on known (input, output) pairs in training set. This is the most common group of algorithms in machine learning algorithms. Classification and regression algorithms are two good examples in this group.

2.1.2.2 Unsupervised learning

In another group of algorithms, the training data consists of only the inputs x and no corresponding outputs. Machine learning algorithms may not be able to predict the output but still extract important information based on the relationships between data points. Algorithms in this group are called unsupervised learning. Algorithms that solve clustering and data dimensionality reduction are typical examples of this group.

2.1.2.3 Semi-supervised learning

The line between supervised and unsupervised learning is sometimes unclear. There are algorithms where the training set consists of pairs (input, output) and other data has only input. These algorithms are called semi-supervised learning.

2.1.2.4 Reinforcement learning

There is another group of machine learning algorithms that may not require training data but rather the model learns to make decisions by communicating with its surroundings. Algorithms in this category continuously make decisions and receive feedback from the environment to reinforce their own behavior. This group of algorithms is called reinforcement learning.

2.2 What is Deep Learning?

Ten years ago, computers would not have been able to categorize thousands of different objects in photographs, write captions for pictures, mimic speech and writing, communicate with people in multiple languages, or even compose poetry and music. However, deep learning has made these tasks possible for computers.

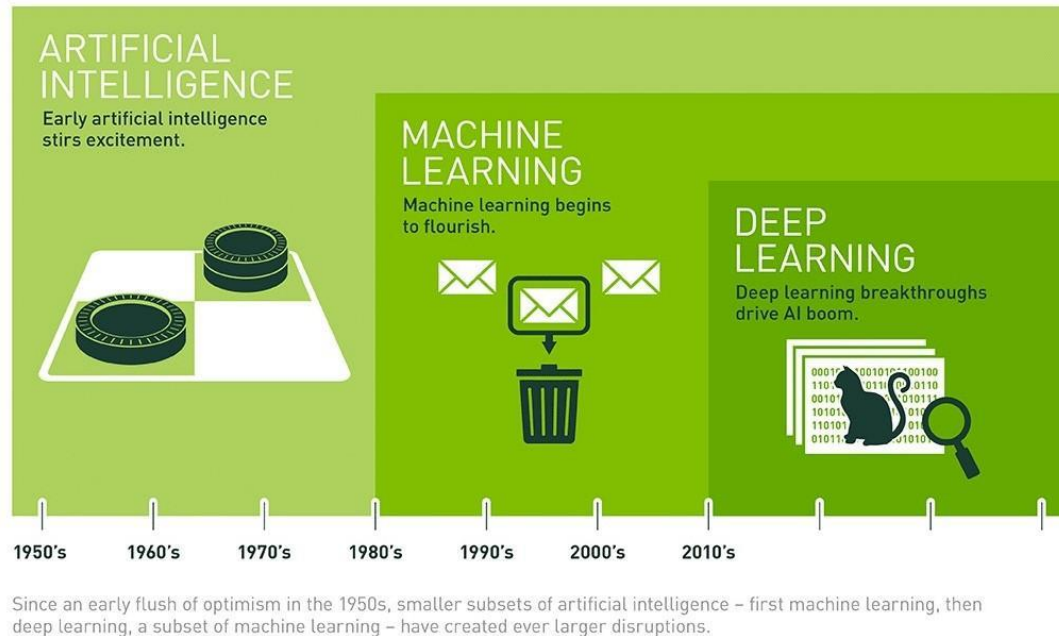


Figure 2: Relationship between AI, ML and DL

We can see that Deep learning is just a small branch of Machine Learning. However, in the past 5 years, Deep Learning has been mentioned a lot as a new trend of the AI revolution. There are several reasons as follows:

- Data explosion: Deep learning exploits Big Data (big data) with much higher accuracy than other Machine Learning methods on data sets, especially for images.
- Development hardware: The arrival of NVIDIA's GTX 10 series GPUs released in 2014 with high computing performance as well as low prices accessible to most people led to the fact that Deep Learning research is no longer available. These are problems that are only studied in expensive laboratories of prestigious universities and large companies.

Some of popular deep learning algorithms are:

- Deep Boltzmann Machine (DBM)
- Deep Belief Networks (DBN)
- Convolutional Neural Network (CNN)
- Stacked Auto-Encoders

2.3 Convolutional Neural Network

2.3.1 Convolution integral

Convolution is visualized by shifting the kernel matrix through all the pixels in the image in turn.

An integral component of Convolutional calculus (convolution) is the kernel matrix (filter). The anchor point of the kernel will determine the corresponding matrix area on the image to

convolution, usually the anchor point is chosen as the center of the kernel. The value of each element on the kernel is considered as a composite factor with each pixel gray value in the region corresponding to the kernel in turn.

The displacement starts from the top left corner of the image. And place the corresponding anchor point at the pixel under consideration. At each displacement, perform a new result calculation for the pixel in question using the convolution formula.

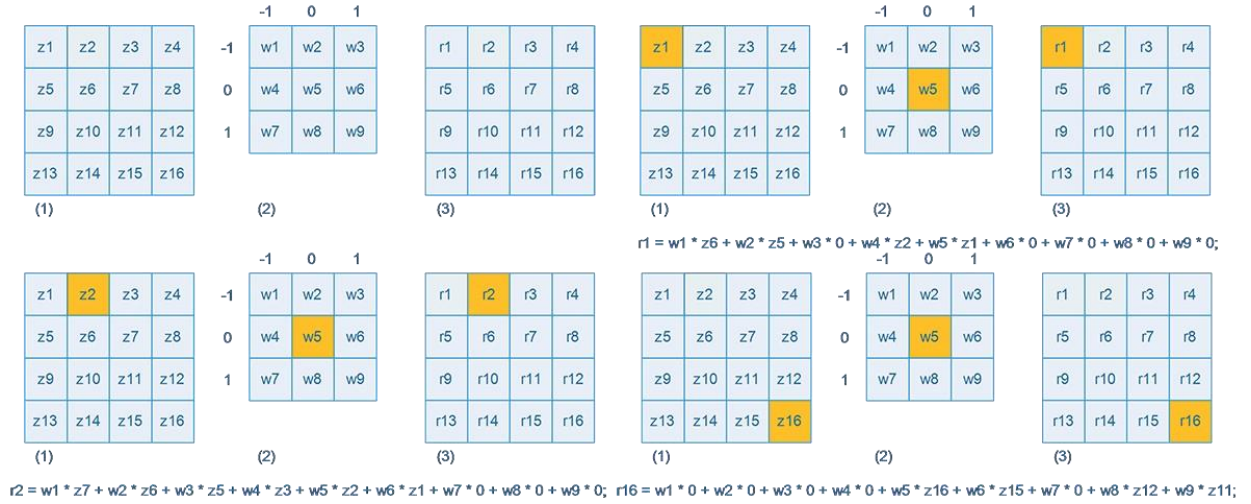


Figure 3: Convolution equation

The purpose of the convolution calculation on the image is to blur and sharpen the image; define paths;... Each different kernel will have different convolution calculations.


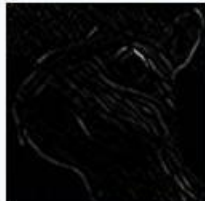



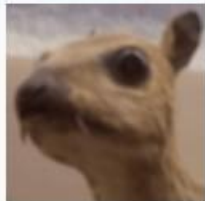
Operation	Kernel ω	Image result $g(x,y)$
Identity	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$	
Edge detection	$\begin{bmatrix} 1 & 0 & -1 \\ 0 & 0 & 0 \\ -1 & 0 & 1 \end{bmatrix}$	
	$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$	
	$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$	
Sharpen	$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$	
Box blur (normalized)	$\frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$	

Figure 4: Example of Kernel

2.3.2 Convolutional Neural Network

A CNN network is a collection of Convolution layers that overlap and use nonlinear activation functions like ReLU and tanh to activate the weights in the nodes. Each class after passing activation functions will generate more abstract information for the next classes.

Each class after passing activation functions will generate more abstract information for the next classes. In the feedforward neural network model, each input neuron gives each output neuron in subsequent layers.

This model is called fully connected layer or fully connected network. In the CNNs model, the opposite is true. The layers are linked together through the convolution mechanism.

The next layer is the convolution result from the previous layer, so we have local connections. Thus, each neuron in the next layer is generated from the result of a filter applied to a local image region of the previous neuron.

Each class that uses different filters usually takes hundreds of thousands of such filters and combines their results. In addition, there are some other layers such as pooling/subsampling layer used to filter out more useful information (remove noise information).

During the training of the network, the CNN automatically learns the values through the filter classes based on how you do it. For example, in the image classification task, CNNs will try to find the optimal parameters for the corresponding filters in the order raw pixels > edges > shapes > facial > high-level features. The last layer is used to layer the image.

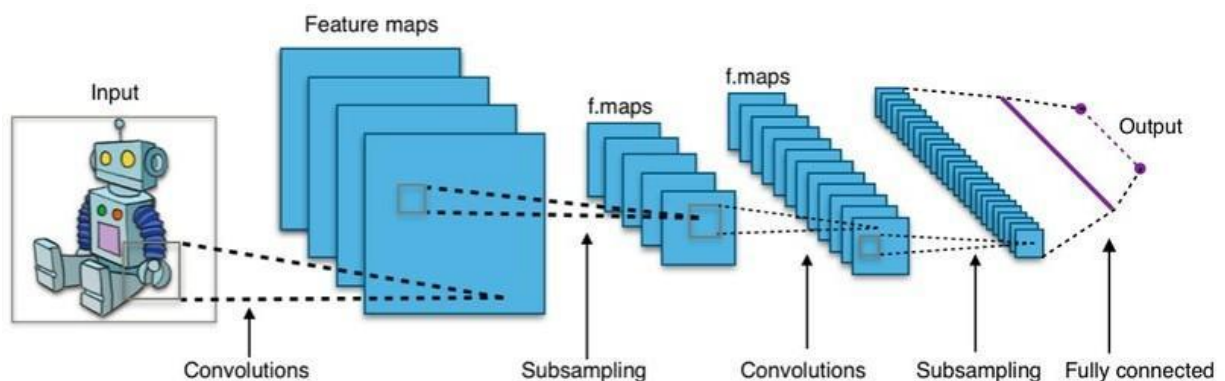


Figure 5: Convolutional Neural Network

2.3.3 Basic classes in CNN

2.3.3.1 Convolutional layer

The class that does all the calculations. CNN uses filters to apply to areas of the image. These filter maps are a 3-dimensional matrix, which includes parameters.

2.3.3.2 Pooling layer

The pooling layer will be placed between the Convolutional layers to reduce the parameters if the input is too large.

There are two common types of pooling layers: max pooling and average pooling.

2.3.3.3 Relu layer

Relu layer is an activation function in neural network. This function has the effect of simulating neurons with a pulse transmission rate through the axon. In the activation function, there are basic functions such as: Sigmoid, Tanh, Relu, Leaky relu, Maxout.

The Relu layer is used after each filter map has been computed and the relu function is applied to all values of the filter map.

2.3.3.4 Fully connected layer

This class is used to output the result. After the Convolutional layer and pooling layer have received the transmitted images, we will get the result that the model has read quite a lot of information about the image. Fully connected layer is used to link those features and output the output.

2.3.4 Structure of CNN

The CNN network consists of many overlapping Convolution layers, using the relu and tanh functions to activate the weights. Each class after activation will produce abstract results for the following classes. Each successive layer represents the result of the previous layer. Through the training process, the CNN layer layers automatically learn the values represented by the filter layers.

The CNN model is invariant and associative. In case the same object is projected at different angles, the accuracy will be affected. For Convolutional, rotation and scaling will use the pooling layer to use as invariants of the other properties. Therefore, CNN gives high accuracy results in models

The basic structure of CNN consists of 3 main parts:

- Local receptive fields are also known as local fields. The effect of this layer is to separate and filter the data and information of the image and select the most valuable image areas to use.
- Shared weights and bias have the main effect of minimizing the number of parameters in the current CNN network. Since each Convolution has different feature maps, each feature map helps to detect some feature in the image.
- The Pooling layer is the composite layer. This is almost the last layer before the results are released. Pooling layer will have the effect of simplifying the output information. Information after being calculated and scanned layers will go to the pooling layer to reduce unnecessary information, then give the best results.

CHAPTER 3: DEEP LEARNING APPLICATION TO SOLVE THE PROBLEM OF CLASSIFICATION OF INTERIOR DESIGN STYLE

3.1 Related works

3.1.1 Research on “Accessing and extracting the characteristics of the interior”

3.1.1.1 Research Summary

This report aims to propose a Deep Learning-based approach to automatically recognize the design features of interior design elements using certain digital images. Recent image recognition techniques using neural networks have shown great success in various research and industrial fields. The open source and pre-trained image recognition models that support the image recognition task allow the team to easily retrain the models to apply them on any platform. The report also describes how to apply those techniques to the interior design process and describes some of the illustrative results in that approach. Furniture is one of the most popular interior design elements whose secondary feature includes implicit design features, such as style, shape, function as well as obvious attributes, such as composition, material and size. This paper shows how to retrain the model to extract some features to effectively manage and use that design information. The target device is the chair, and the targeted design features are limited to functional features, materials, capacity, and design style. A total of 3933 chair image datasets and 6 retrained image recognition models were used for retraining. Through the combination of these multiple models, an inferential performance has also been described.

3.1.1.2 Results of the research

This research work aims to develop automatic extraction of the entire interior design equipment and its design features from images and use it for interior design. Among the research needed for that, the detection of interior elements was possible and showed a high degree of accuracy. As a baseline study for the next step, this paper describes just the necessary techniques and approach surveys and uses them to test to retrain the deep learning model to recognize the design devices. interior and its design features.

The team examined available techniques such as CNNs and image recognition libraries and used them for recognition purposes. In the report, interior design devices identified are limited to devices such as: seats such as chairs, sofas and stools. Design features are limited to a few visually identifiable features on the image. Training image data is collected from Google image search and relevant professional website like Houzz, one of the most popular furniture e-commerce website. The process of this paper is summarized as follows:

- Automatic training and extraction of chair design features: The furniture and its features are identified and interesting criteria characteristics can be visually inspected. Next, the image dataset was built using features from multiple web-based sources. Multiple image recognition models are retrained for each type of furniture.

- Automation and some of its design features using retraining models: integration of multiple automation mining models, different design features inferred from the given image.

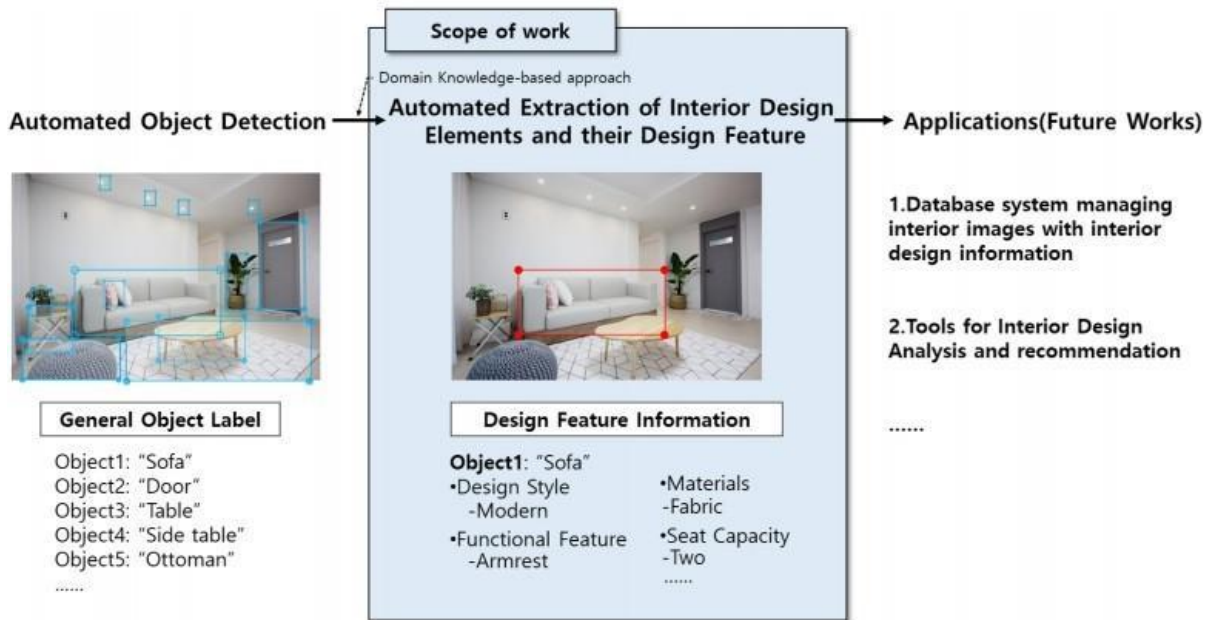


Figure 6: The scope of work automatically exploits the devices in interior design and their design features.

3.1.2 Research on “Multimodal search engine for fashion and interior design”

3.1.2.1 Research Summary

In the paper, the team proposed a multimodal search engine that combines visual and textual features to retrieve items from a query-like multimedia database. The goal of the team's application is to enable visual retrieval of fashion items such as clothing or furniture. Existing search engines only consider text input as an additional source of information about the query image and do not correspond to a real-life situation where a user searches for “same t-shirt but in cotton fabric” “. The team's new method, called DeepStyle, mitigates those shortcomings by using a common neural network architecture to model contextual dependencies between the features of different methods. . The team demonstrated the robustness of this method on two different challenging data sets of fashion and interior items, where our DeepStyle engine outperformed the basic methods by more than 20%. on the tested data sets. Our search engine is commercially deployed and available through a Web-based application.

3.1.2.2 Results of the research

In this paper, we propose a new method for multi-method query. The proposed method is a Siamese neural network architecture that learns stylistic similarity by leveraging empirical context information - how often the given items appear in the same stylistic context. The

research's method goes beyond basic methods and achieves better results to create stylistically compatible groups of items.

The biggest advantage of this method is that it is twice as efficient. First, it allows extending the visual query with arbitrary text input and conveys information not present in the visual input, thus allowing users to find more relevant products. Second, it retrieves stylistically similar results.

The main disadvantage of this approach is the need for large labeled data about the contextual image (the contextual information where the items appear together). Semi-supervised learning methods that can reduce the need for such data are up to our future work.

The study successfully applied the methodology to several commercial domain applications - fashion and interior design, by mining product images and their associated metadata. Finally, the team implemented a smart pre-existing web implementation of our product and released a new dataset with IKEA furniture items.

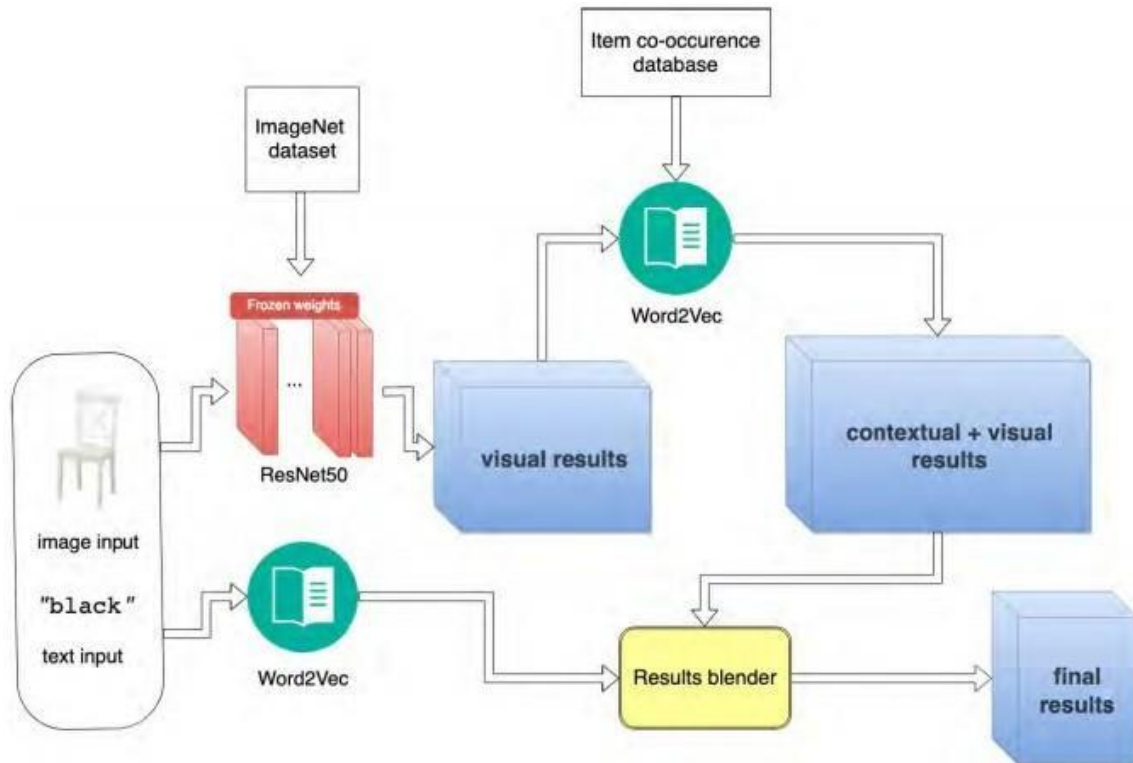


Figure 7: A high-level overview of the Early-fusion Blending architecture

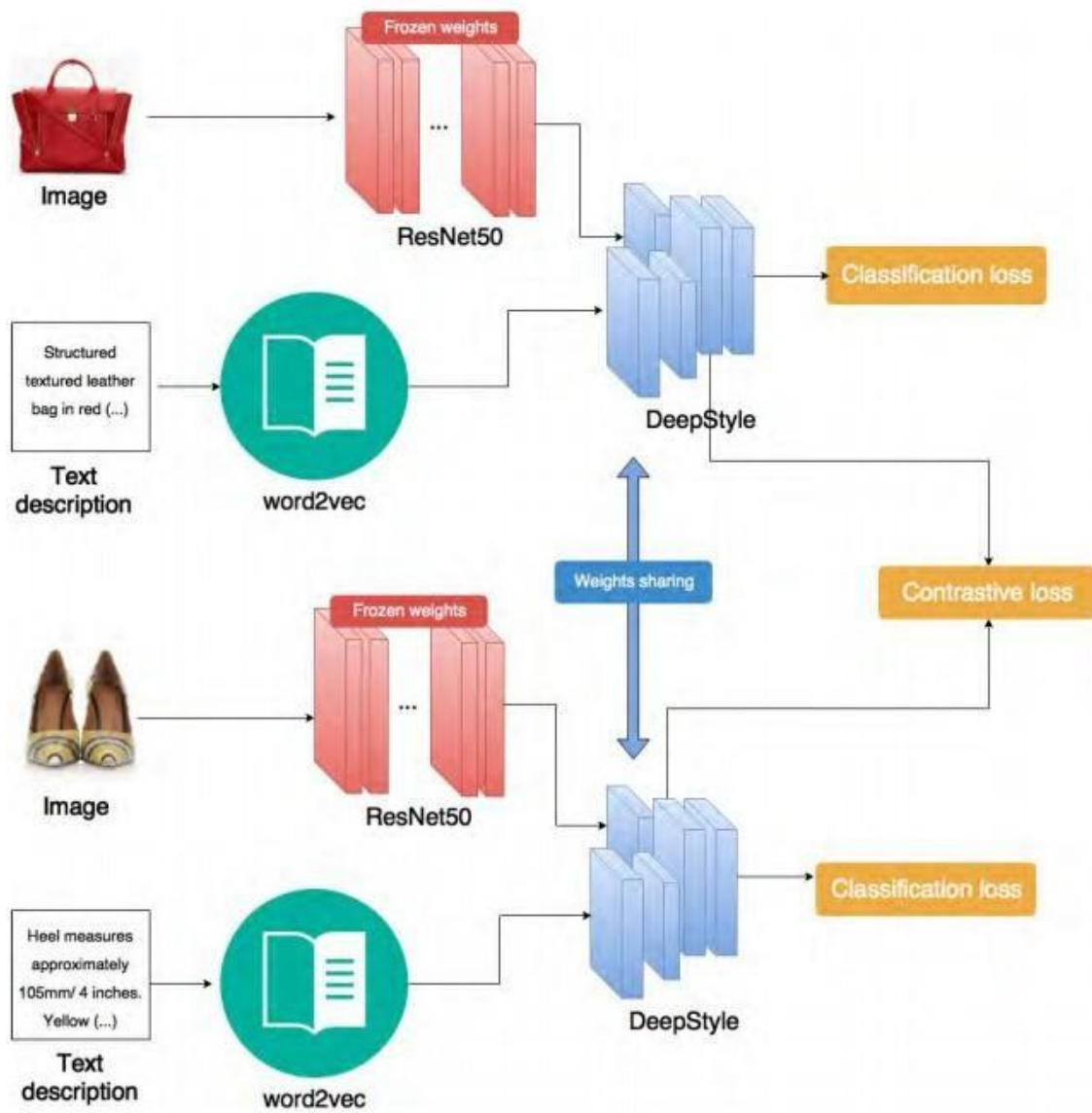


Figure 8: Architecture of the team's DeepStyle-Siamese network

3.2 Suggested method

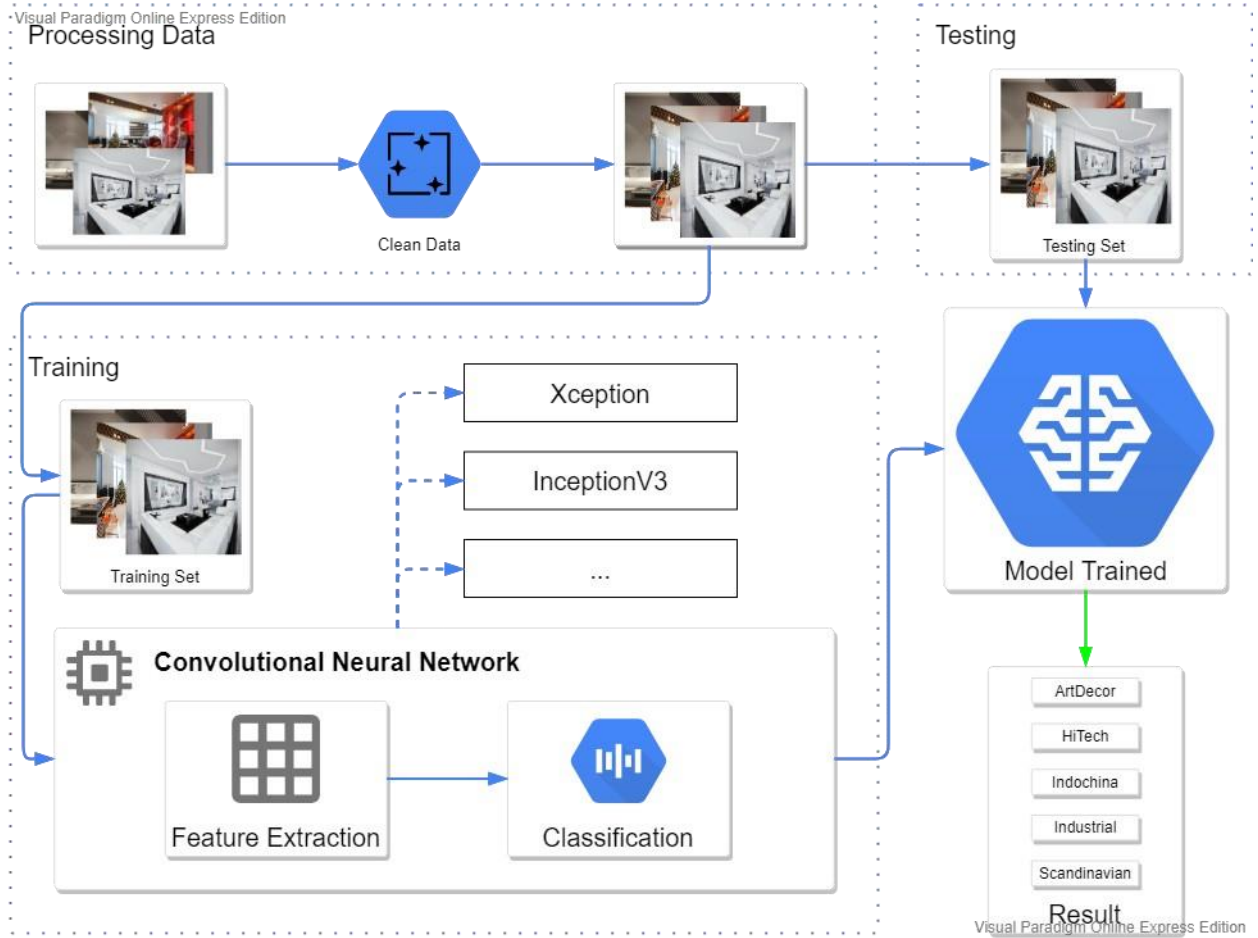


Figure 9: An overview of the proposed method.

3.3 Models

3.3.1 VGG19

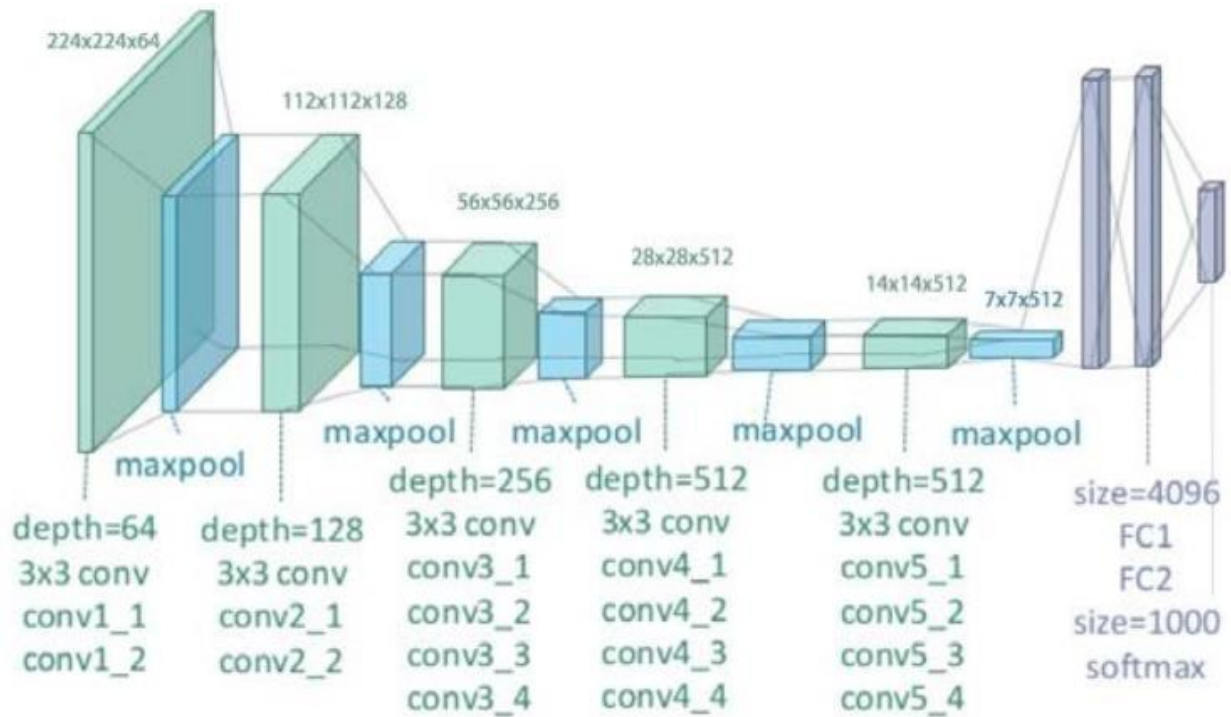


Figure 10: VGG19 architecture

VGG-19 is a convolutional neural network that is 19 layers deep. You can load a pretrained version of the network trained on more than a million images from the ImageNet database. The pretrained network can classify images into 1000 object categories, such as keyboard, mouse, pencil, and many animals. As a result, the network has learned rich feature representations for a wide range of images. The network has an image input size of 224-by-224.

3.3.2 InceptionV3

Inception-V3 is the successor of Inception-V1 which includes 24 million parameters. All Inception-V3's convolution layers are followed by a batch normalization layer and a ReLU activation. Batch normalization is a technique to normalize the input by minibatch at each layer according to the normal distribution, making the algorithm training process faster.

Inception-V3 solves the problem of representational bottlenecks. That is, the size of the output is suddenly reduced compared to the input and there is a more efficient way of computing by using factorisation methods.

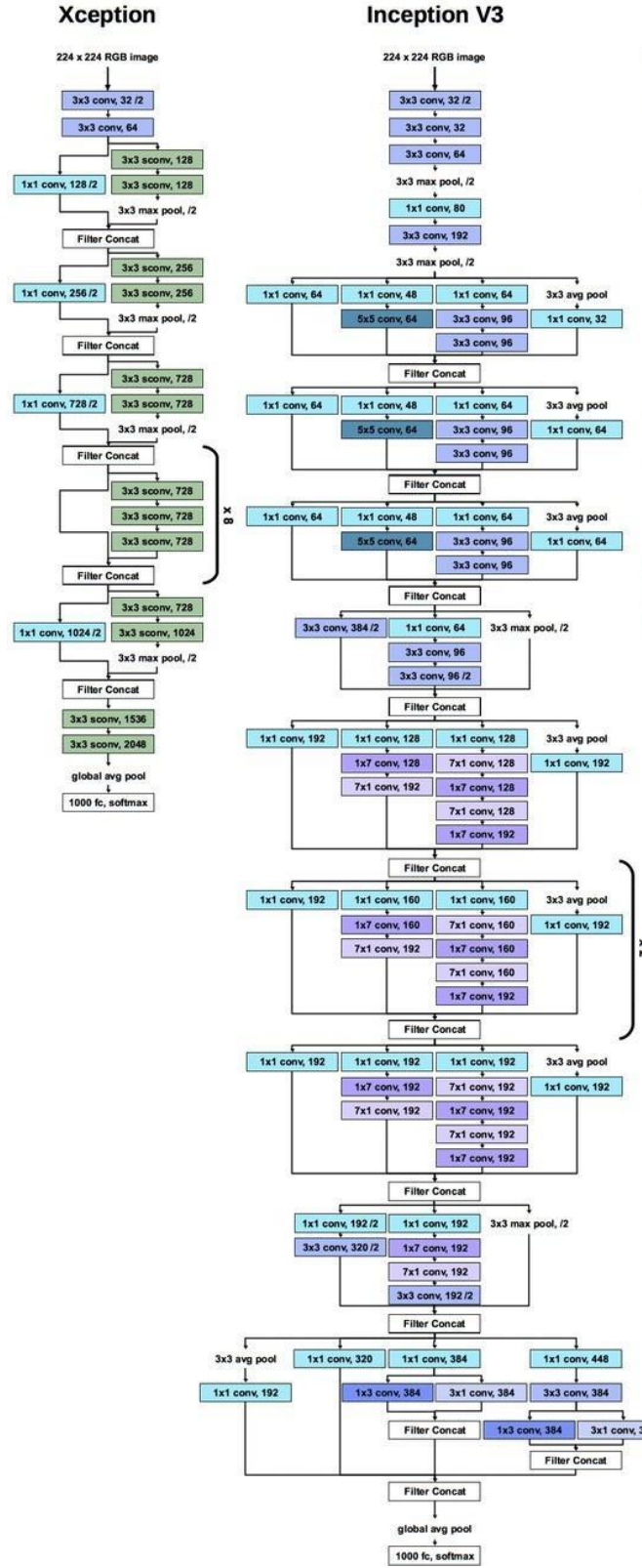


Figure 11: Exception and Inception V3 architectures

3.3.3 Xception

Xception stands for "Extreme Inception" which is a more powerful version of the Inception architecture. The Xception architecture has 36 layers of integration that form the basis of the network's feature extraction. The 36 convolutional layers are structured into 14 modules, all of which have linear residual connections around them, except for the first and last module. In a nutshell, the Xception architecture is a linear stack of depth-separable convolutional layers with the remaining connections. This makes the architecture very easy to define and modify.

3.3.4 Resnet50

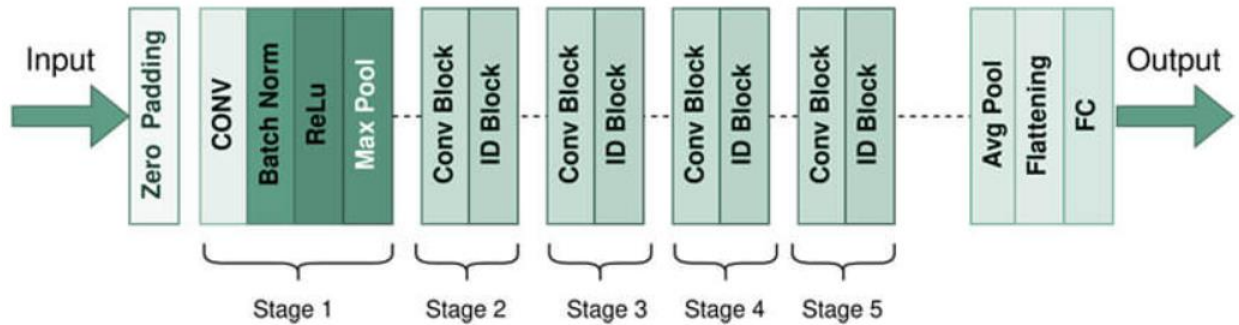


Figure 12: Resnet50 architecture

ResNet stands for Residual Network and is a specific type of convolutional neural network (CNN) introduced in the 2015 paper "Deep Residual Learning for Image Recognition" by He Kaiming, Zhang Xiangyu, Ren Shaoqing, and Sun Jian. CNNs are commonly used to power computer vision applications.

ResNet-50 is a 50-layer convolutional neural network (48 convolutional layers, one MaxPool layer, and one average pool layer). The network has an image input size of 224-by-224.

CHAPTER 4: EXPERIMENTS AND RESULTS

4.1 Dataset

4.1.1 Reference source for data

- Some sources of free and high-quality 2D images:
- Image search engine - Google Image.
- Website specializing in sharing stock photos under license - Unsplash.
- Website to share photos in the form of social networks, post and classify as photo stickers - Pinterest.
- There are also some other photo websites such as: Interior Design Magazine, Interior Design Ideas & Home Decorating Inspiration, etc.

4.1.2 Data collection

4.1.2.1 Why is data collection important?

Data collecting enables us to understand the distinctive characteristics of certain information so that we may evaluate the data using recurring patterns. By applying machine learning algorithms to create prediction models from such patterns, we may find trends and forecast changes in the future.

The technique used to gather high-quality data is essential to creating performance models with strong predictive accuracy since predictive models are only as good as the data on which they are based. Errors, abnormalities, and information pertinent to the intended prediction must all be included in the data. For instance, knowledge of the area's location, direction, and population density will be highly helpful when determining house costs. Since the weather has little bearing on housing prices, we won't need to include it in the list. gathered facts.

4.1.2.2 Data collection operations

The team utilized a data set that was already accessible and supplied by supervisor, and we increased the data set by the number of samples of one label since it had a lot of variations and was too small. The following are some of the strategies the team used to locate new data:

- Automatically crop objects from each single image with YOLOv5.
- Manually crop objects with care.

4.1.2.3 Prepare, Explore, Clean, and Consistency of Data

Following steps, in order:

Investigate, quantify, and evaluate the quality of the data.

Training set and test/validation set should be separated from the entire data set.

Data homogenization and picture data reading.

- Use tensorflow's available function to read images from directories
- Make the picture 224x224 or 299x299 in overall size.

- Labels should also be converted to their ordinal places.

Recommended Code:

Define Data Path

```
train_path = r'./Data/train'  
test_path = r'./Data/test'  
val_path = r'./Data/val'
```

Define Image Input Size

```
IMG_SIZE = 299
```

Figure 13: PATH of application

Load Data From Directory

```
batch_size_train = 100  
batch_size_val = 100
```

```
training_datagen = ImageDataGenerator(preprocessing_function=preprocess_input)  
  
# validation_datagen = ImageDataGenerator(rescale = 1./255)  
validation_datagen = ImageDataGenerator(preprocessing_function=preprocess_input)  
  
test_datagen = ImageDataGenerator(rescale = 1./255)  
  
train_generator = training_datagen.flow_from_directory(  
    train_path,  
    target_size=(IMG_SIZE,IMG_SIZE),  
    class_mode='categorical',  
    batch_size=batch_size_train  
)  
  
validation_generator = validation_datagen.flow_from_directory(  
    val_path,  
    target_size=(IMG_SIZE,IMG_SIZE),  
    class_mode='categorical',  
    batch_size=batch_size_val  
)  
  
test_generator = test_datagen.flow_from_directory(  
    test_path,  
    target_size=(IMG_SIZE,IMG_SIZE),  
    class_mode='categorical',  
    batch_size=batch_size_val  
)
```

Figure 14: Load data from directory

After the above steps, 3 DirectoryIterator will be created. train_generator and validation_generator will be passed through the model for training, test_generator will be used for final evaluation.

4.1.3 Description of the dataset

- The dataset includes all 3056 original images and 11432 objects after cropping and filtering
- Asynchronous size (not the same general size).
- The image includes 5 types of labels: ArtDecor, HiTech, Indochina, Industrial and Scandinavian.
- The number of images for each label type is not uniform.
- Same image format *.jpg.
- The following table shows the number of images for each type of label:

Design Style	No Original Images	No Crop Objects	Image Size (smallest)	Image Size (largest)
ArtDecor	601	2336	216x42	3982x3830
Hitech	597	2213	109x49	5754x7987
Indochina	718	2376	81x153	3115x1913
Industrial	591	2343	49x85	2915x1724
Scandinavian	549	2164	99x100	3956x2633

Table 1: Data statistic

4.2 Environment and Tools

4.2.1 Python

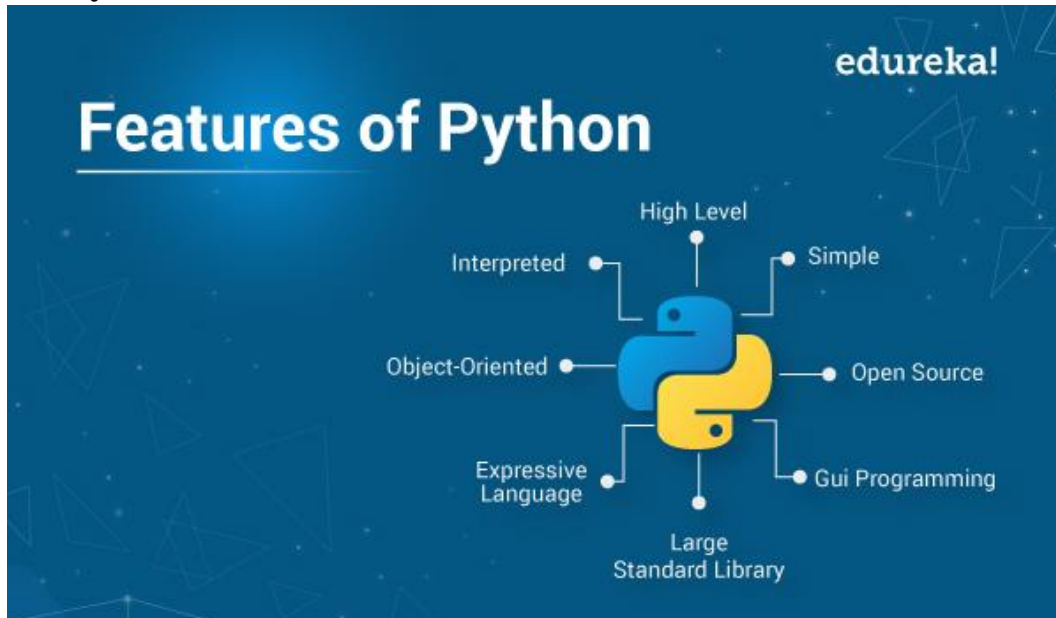


Figure 15: Python feature

Python is a high-level, general-purpose programming language. Its design philosophy emphasizes code readability with the use of significant indentation.

Python is dynamically typed and garbage-collected. It supports multiple programming paradigms, including structured (particularly procedural), object-oriented and functional programming. It is often described as a "batteries included" language due to its comprehensive standard library.

4.2.2 Google Colab



Figure 16: Google colab

Colaboratory, or “Colab” for short, is a product from Google Research. Colab allows anybody to write and execute arbitrary python code through the browser, and is especially well

suited to machine learning, data analysis and education. More technically, Colab is a hosted Jupyter notebook service that requires no setup to use, while providing access free of charge to computing resources including GPUs.

Colab is free of charge to use.

4.2.3 Jupyter



Figure 17: Jupyter

Project Jupyter aims to create open-source software, open standards, and services for interactive computing across many programming languages.

The term "Jupyter Notebook" can be used to refer to either the user-facing program used to edit code and text, or the underlying file format that can be used with a variety of implementations.

The web-based interactive computational environment Jupyter Notebook (formerly known as IPython Notebook) is used to create notebook papers. Several open-source libraries are used in the construction of Jupyter Notebook, including IPython, ZeroMQ, Tornado, jQuery, Bootstrap, and MathJax. A Jupyter Notebook application is a web-based REPL that includes an ordered set of input/output cells that can include code, text (written in Markdown), arithmetic, graphs, and rich media

4.2.4 Flask



Figure 18: Flask

Flask is a micro web framework written in Python. It is classified as a microframework because it does not require particular tools or libraries. It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions. However, Flask supports extensions that can add application features as if they were implemented in Flask itself. Extensions exist for object-relational mappers, form validation, upload handling, various open authentication technologies and several common framework related tools.

4.2.5 ReactJS

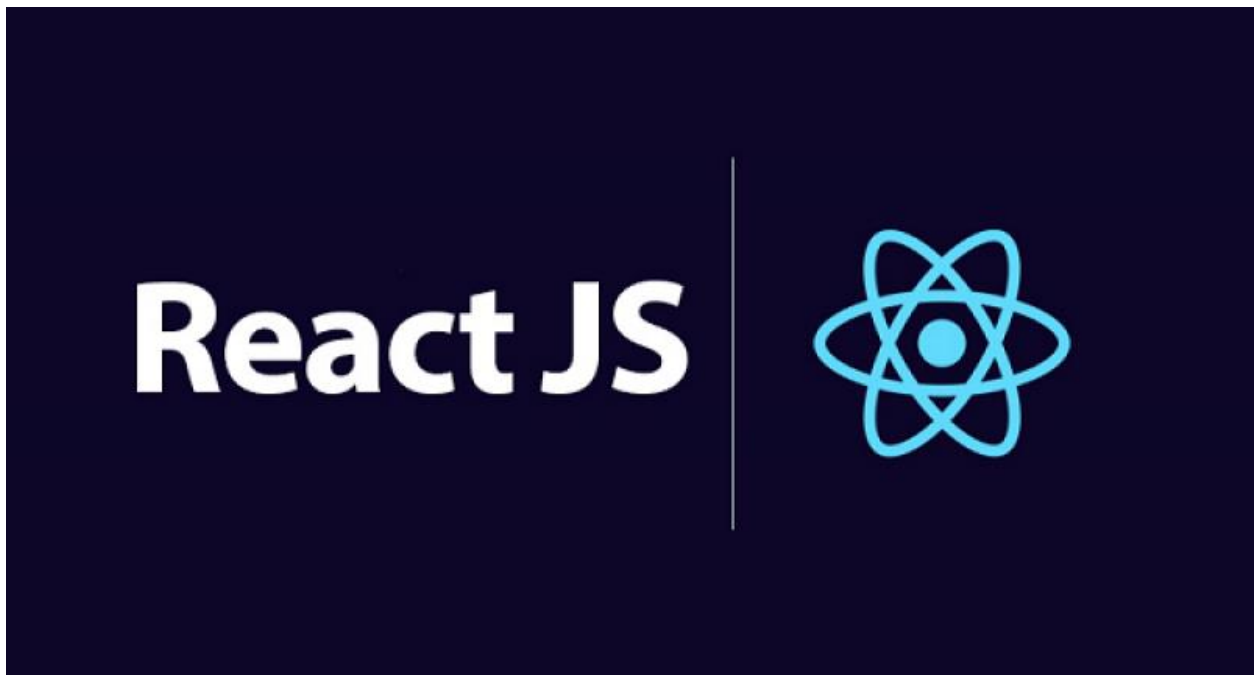


Figure 19: ReactJS

ReactJS is a free and open-source front-end JavaScript library for building user interfaces based on UI components. It is maintained by Meta (formerly Facebook) and a community of individual developers and companies. React can be used as a base in the development of single-page, mobile, or server-rendered applications with frameworks like Next.js. However, React is only concerned with state management and rendering that state to the DOM, so creating React applications usually requires the use of additional libraries for routing, as well as certain client-side functionality.

4.2.6 YOLOv5

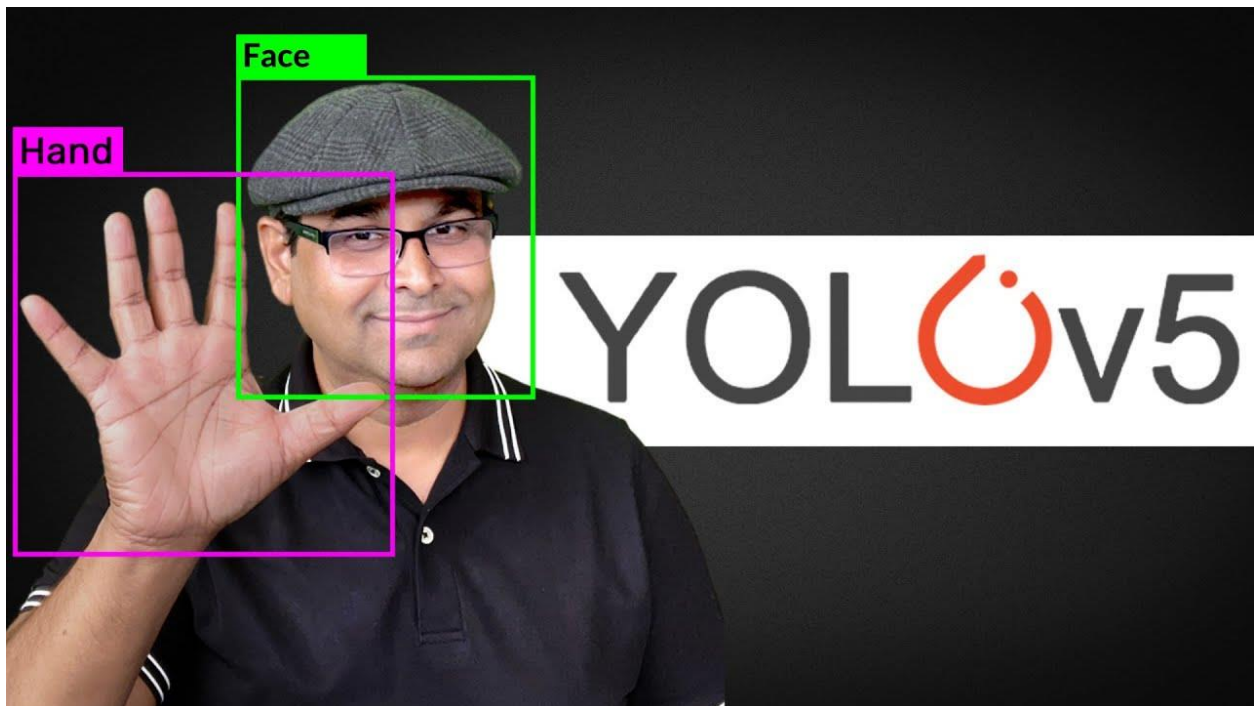


Figure 20: YOLOv5

YOLOv5 is the world's most loved vision AI, representing Ultralytics open-source research into future vision AI methods, incorporating lessons learned and best practices evolved over thousands of hours of research and development.

We use this library to detect interior objects in the image and thereby identify the style of those furniture objects.

4.3 Implementation process

4.3.1 Choose the train model

The purpose of the project is to apply deep learning to extract features of images to classify images and one of the advanced Deep Learning models is Convolutional Neural Network can help us to do this.

Keras is an open source software library that provides a Python interface to artificial neural networks. Keras also provides pre-built models for learning, research and extension needs.

Pre-built models provided by Keras:

Model	Size(MB)	Train Accuracy	Validation Accuracy	Parameters
<u>Xception</u>	88 MB	0.889	0.571	22,910,480
<u>VGG16</u>	528 MB	-	-	138,357,544
<u>VGG19</u>	549 MB	0.889	0.546	143,667,240
<u>ResNet50</u>	98 MB	0.920	0.606	25,636,712
<u>ResNet101</u>	171 MB	-	-	44,707,176
<u>ResNet152</u>	232 MB	-	-	60,419,944
<u>ResNet50V2</u>	98 MB	-	-	25,613,800
<u>ResNet101V2</u>	171 MB	-	-	44,675,560
<u>ResNet152V2</u>	232 MB	-	-	60,380,648
<u>InceptionV3</u>	92 MB	0.877	0.566	23,851,784
<u>InceptionResNetV2</u>	215 MB	-	-	55,873,736
<u>MobileNet</u>	16 MB	-	-	4,253,864
<u>MobileNetV2</u>	14 MB	-	-	3,538,984
<u>DenseNet121</u>	33 MB	-	-	8,062,504
<u>DenseNet169</u>	57 MB	-	-	14,307,880
<u>DenseNet201</u>	80 MB	-	-	20,242,984
<u>NASNetMobile</u>	23 MB	-	-	5,326,716
<u>NASNetLarge</u>	343 MB			88,949,818
<u>EfficientNetB0</u>	29 MB	-	-	5,330,571
<u>EfficientNetB1</u>	31 MB	-	-	7,856,239
<u>EfficientNetB2</u>	36 MB	-	-	9,177,569
<u>EfficientNetB3</u>	48 MB	-	-	12,320,535
<u>EfficientNetB4</u>	75 MB	-	-	19,466,823
<u>EfficientNetB5</u>	118 MB	-	-	30,562,527
<u>EfficientNetB6</u>	166 MB	-	-	43,265,143
<u>EfficientNetB7</u>	256 MB	-	-	66,658,687

Table 2: Pre-built models provided by Keras

In this research, the team will focus on training on popular pre-built CNN models provided by Keras and test the fit of the models.

Some of the models that our group will do the training:

- VGG19.
- ResNet50.

- InceptionV3.
- Xception.

In the next step, the team will dive into training the model.

4.3.2 Train the selected model with the dataset

4.3.2.1 Keras's pre-built models

When we want to use the model, we use the library of Keras. Using the syntax:
`tf.keras.applications.<model name>(<optional parameters>).`

Where <model name> includes: Xception, VGG16, VGG19, ResNet50, ResNet101, ResNet152, ResNet50V2, ResNet101V2, ResNet152V2, InceptionV3, InceptionResNetV2, MobileNet, MobileNetV2, DenseNet121, DenseNet169, DenseNet201, NASNetMobile, NASNetLarge, EfficientNetB0, EfficientNetB1, EfficientNetB2, EfficientNetB3, EfficientNetB4, EfficientNetB5, EfficientNetB6, EfficientNetB7.

Where <optional parameters> has parameters like:

`include_top`: The boolean value used to accept the fully-connected layer or not, the default will be True. We will pre-train to give another number of layers, so during training we will recreate this layer for the model we need to train, we set this parameter value as False.

`weights`: There will be 2 options: "None" and "imagenet", when choosing "None" then the weights of your model will give random values, and with the "imagenet" option, our model will take the weights of the previously trained model available. In this project, the team will take the previous weighted value of "imagenet" to retrain the model.

`input_shape`: optional shape tuple, only to be specified if `include_top` is False (otherwise the input shape has to be (224, 224, 3) (with `channels_last` data format) or (3, 224, 224) (with `channels_first` data format). We choose (224, 224, 3) and (299, 299, 3) as the values for this parameter since they are recommended for the selected models.

`classes`: optional number of classes to classify images into, only to be specified if `include_top` is True, and if no `weights` argument is specified.

And some other parameters depending on the model. And in this project we leave them at default.

4.3.2.2 Configure for the model

Since we do not apply transfer-learning for our model, we freeze all parameters of precede layers. Then we create a new output layer for our model. Finally, we configure the model with `model.compile()` function, in compile function there are some parameters we need to define:

`optimizer`: Algorithms or methods used to change the attributes of the neural network such as weights and learning rate in order to reduce the losses. Optimizers help to get results faster.

loss: Functions used to determine the error (or loss) between the output of our algorithms and the given target value.

metrics: List of metrics used to monitor and measure the performance of a model (during training and testing)

There are still other parameters but we leave them as default.

Recommended code:

Xception Model

```
xception = Xception(input_shape = (299, 299, 3), weights = 'imagenet', include_top = False)

for layer in xception.layers:
    layer.trainable = False

x = Flatten()(xception.output)
prediction = Dense(5, activation='softmax')(x)

model = Model(inputs=xception.input, outputs=prediction)
model.summary()
model.compile(loss='categorical_crossentropy', optimizer="adam", metrics=['accuracy'])
```

Figure 21: Xception Model

4.3.2.3 Train the model

After everything is set up, now it's time to train our model with the train and validation set, which were prepared before.

For this, we use fit() function, which was provided by Tensorflow. fit() function allow us to train the model for a fixed number of epochs (iterations on a dataset)

Recommend code:

```
history = model.fit(train_generator, validation_data = validation_generator
                    , epochs = 20, steps_per_epoch = 20, shuffle=True)
```

Figure 22: Code for train model

4.3.2.4 Evaluate the model

Information about loss and accuracy is shown, with the two values plotted on a graph in terms of epochs.

Recommended code:

Train accuracy and validation accuracy


```
plt.figure(figsize=(10, 8))
plt.plot(history.history['accuracy'], label='train acc')
plt.plot(history.history['val_accuracy'], label='val acc')
plt.legend()
plt.title('Accuracy')
plt.show()
```

Figure 23: Code for plot accuracy

Train loss and validation loss

```
plt.figure(figsize=(10, 8))
plt.plot(history.history['loss'], label='train loss')
plt.plot(history.history['val_loss'], label='val loss')
plt.legend()
plt.title('Loss')
plt.show()
```

Figure 24: Code for plot loss

Result: Xception model's train (validation) loss and train (validation) accuracy information graph.

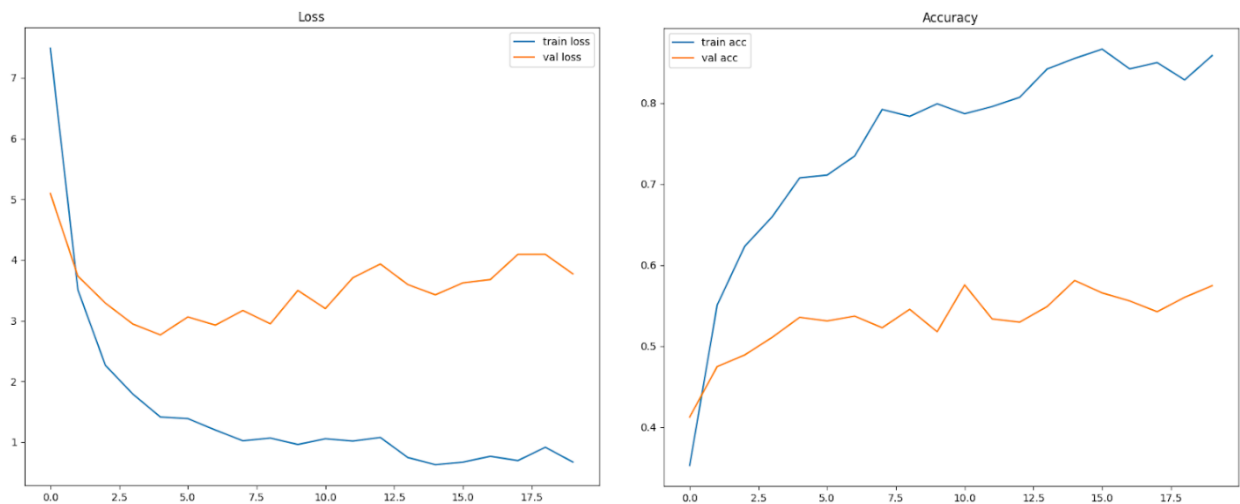


Figure 25: Loss and accuracy graph

Display confusion matrix.

Recommended code:

Result: Confusion matrix of Xception model.

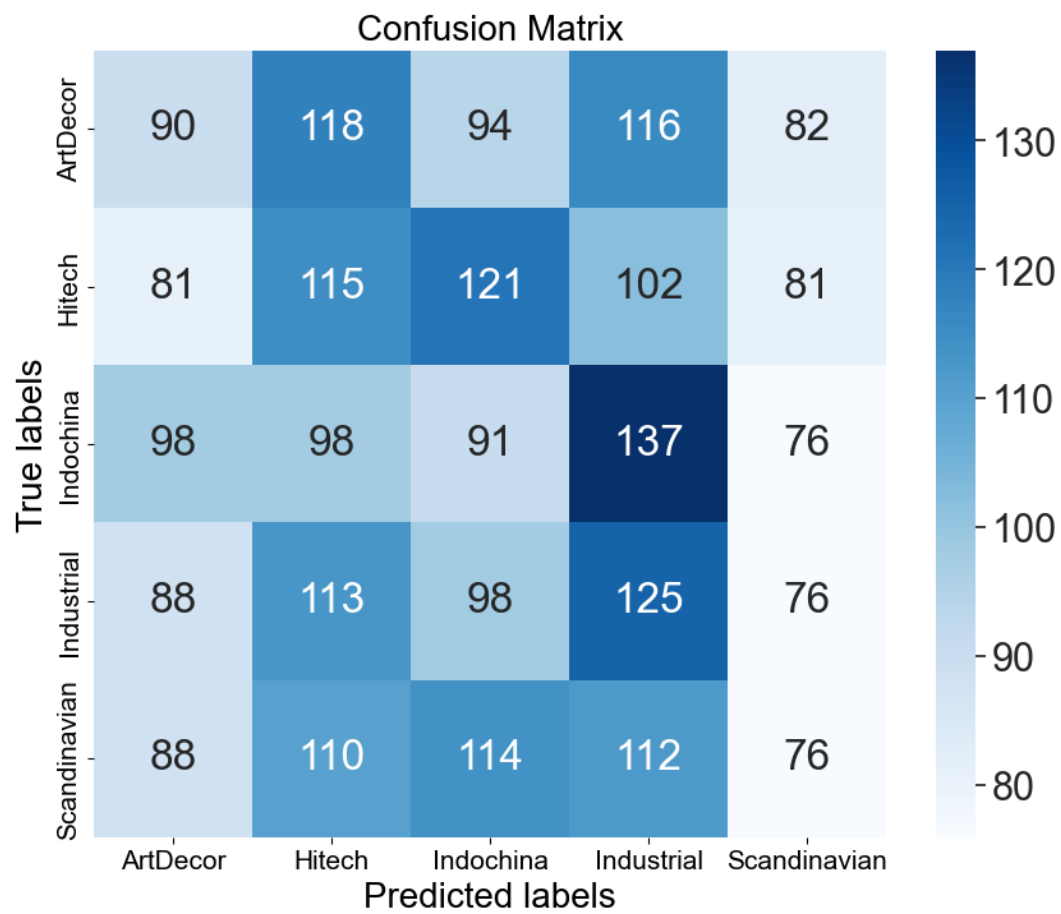


Figure 26: Confusion matrix

4.3.3 Web application development

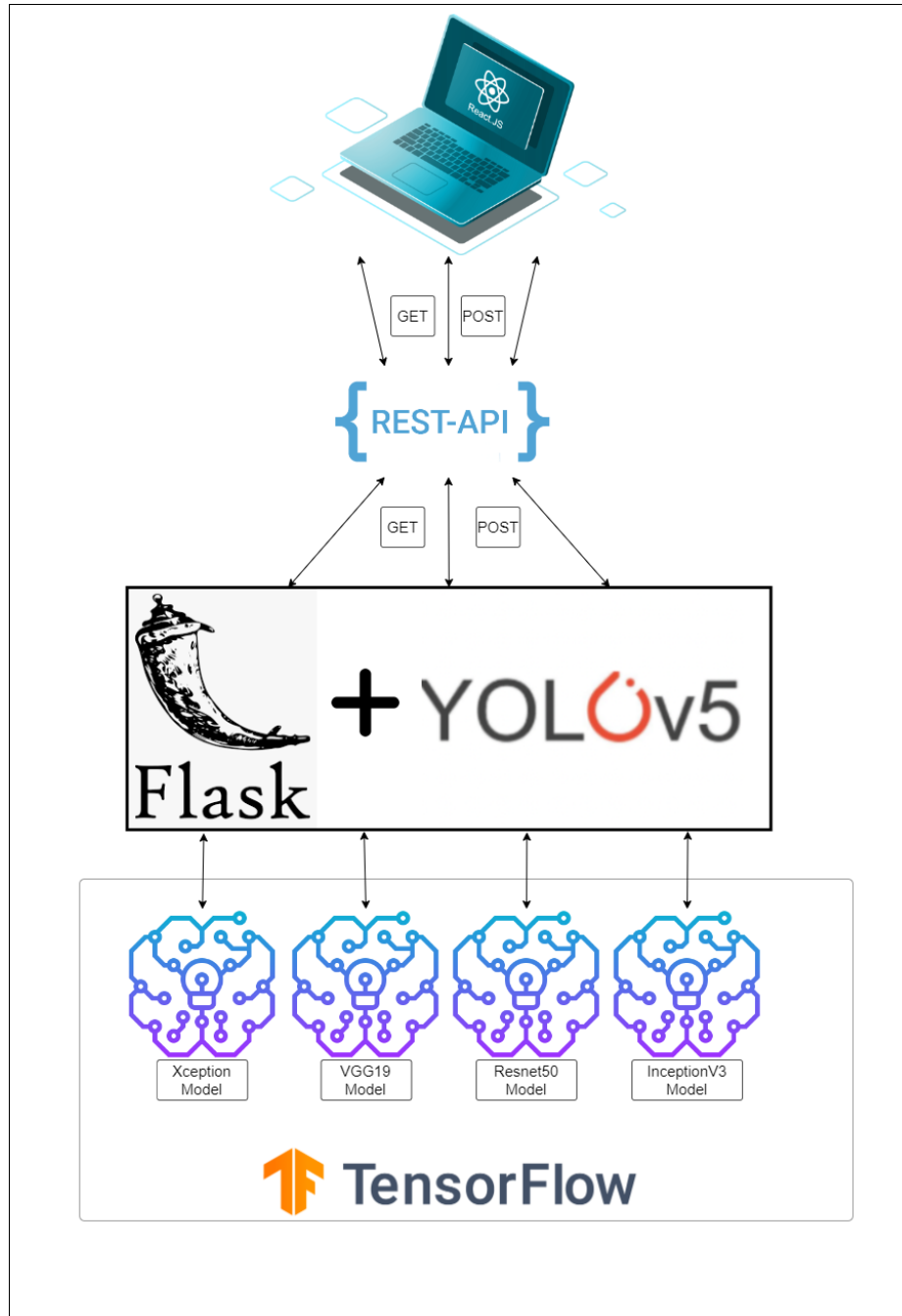


Figure 27: System proceed overview

3.3.1 Create APIs with high-accuracy models.

After comparing the results of all models after training, we can draw the 4 models with the highest prediction accuracy rate:

InceptionV3	0.87%
Resnet50	0.92%
VGG19	0.88%
Xception	0.88%

After the models have been established and built, we will utilize the application to build integrated APIs that can be used on a variety of platforms. To integrate the application for classifying interior design styles, we will build a web server with an API gateway, a web app for dispersing the API, and a basic website for classifying interior design styles.

- Provides the following APIs:
- GET method – “https://<domain>/”: Will return a simple website interface that allows users to upload images and predict what kind of design style. And the website has the following interface:

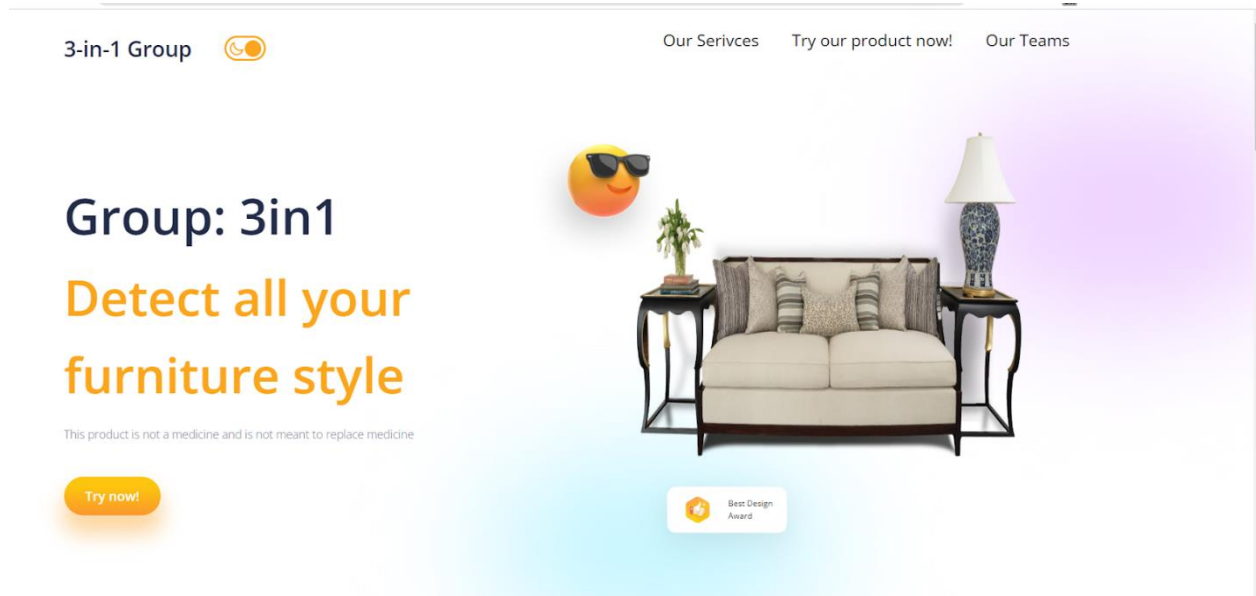


Figure 28: Welcome screen of the application

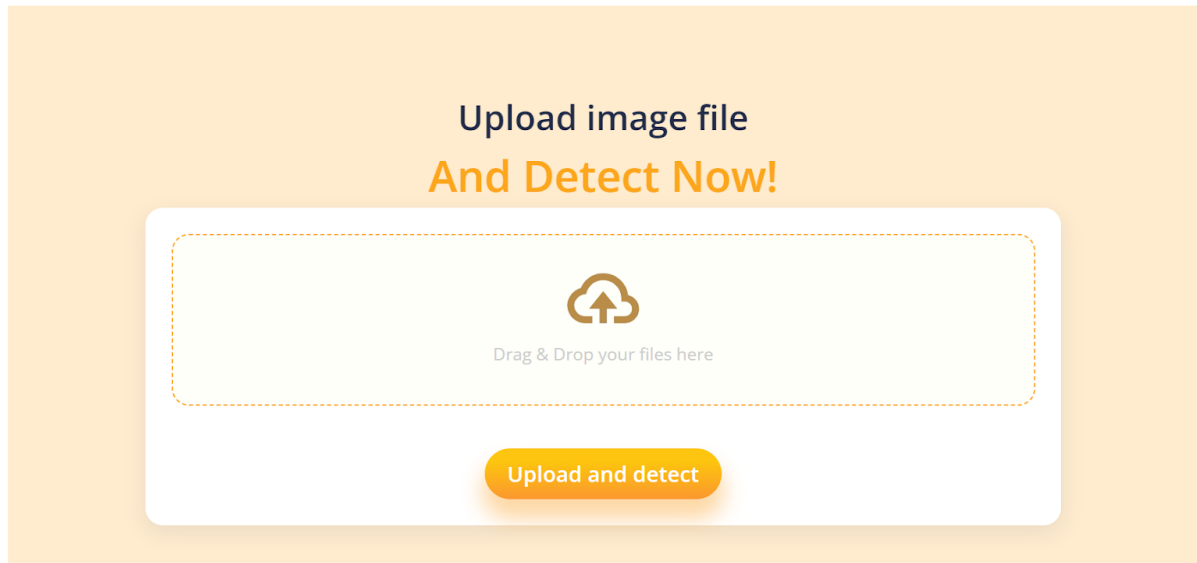


Figure 29: Main screen of the application

POST method – “https://<domain>/”: The method will request an additional image file. The returned result is json with the following components: Prediction results of 3 types of models and how much each label is, success or failure status and labels.

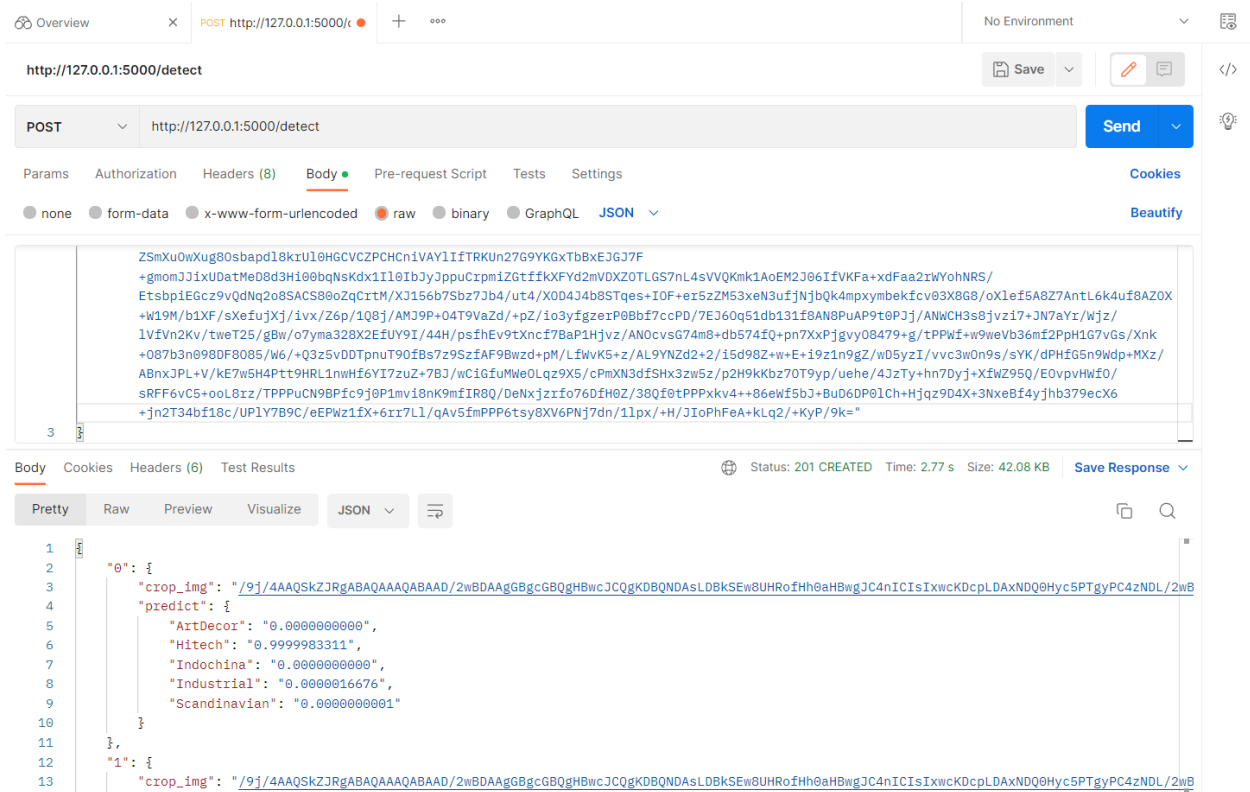


Figure 30: Test API with Postman

3.3.2 Create application interface using ReactJS

We now have a straightforward API for classifying design trends. The next objective is to create a website interface, which is a more practical application that can reach people as soon as feasible. The team learnt about ReactJS, a pretty well-known framework, to construct the website interface. ReactJS is a front-end JavaScript toolkit for creating user interfaces based on UI components that is open-source and free. It is kept up-to-date by Meta and a group of independent programmers and businesses.

It was difficult for the team to build an app on an outdated but often updated platform in such a short amount of time, but we succeeded in doing so and included functionality like the following:

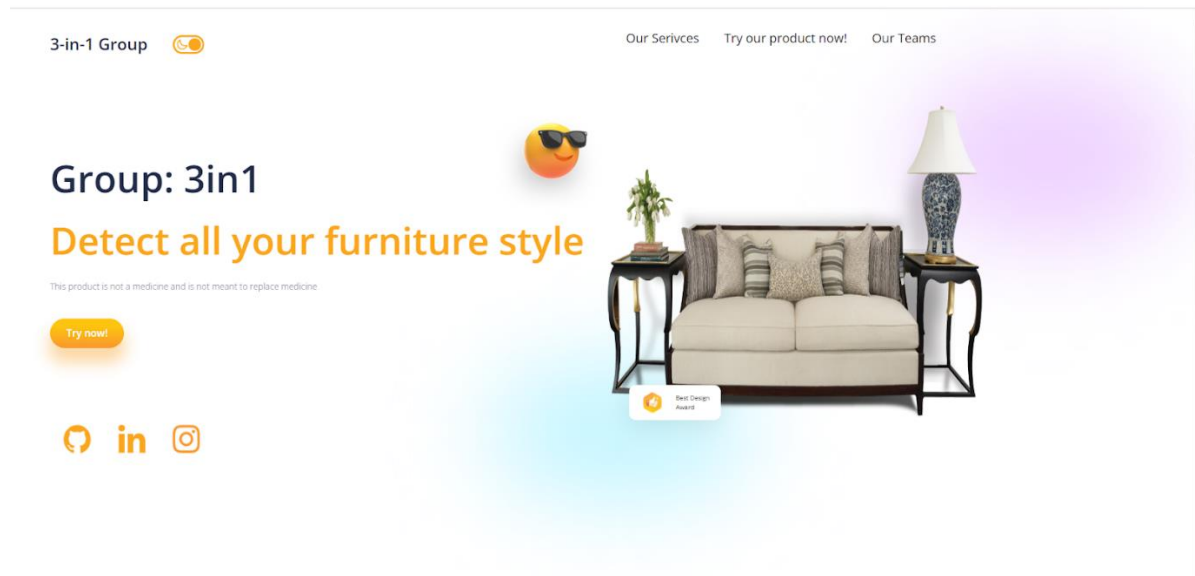


Figure 31: Main screen created by ReactJS

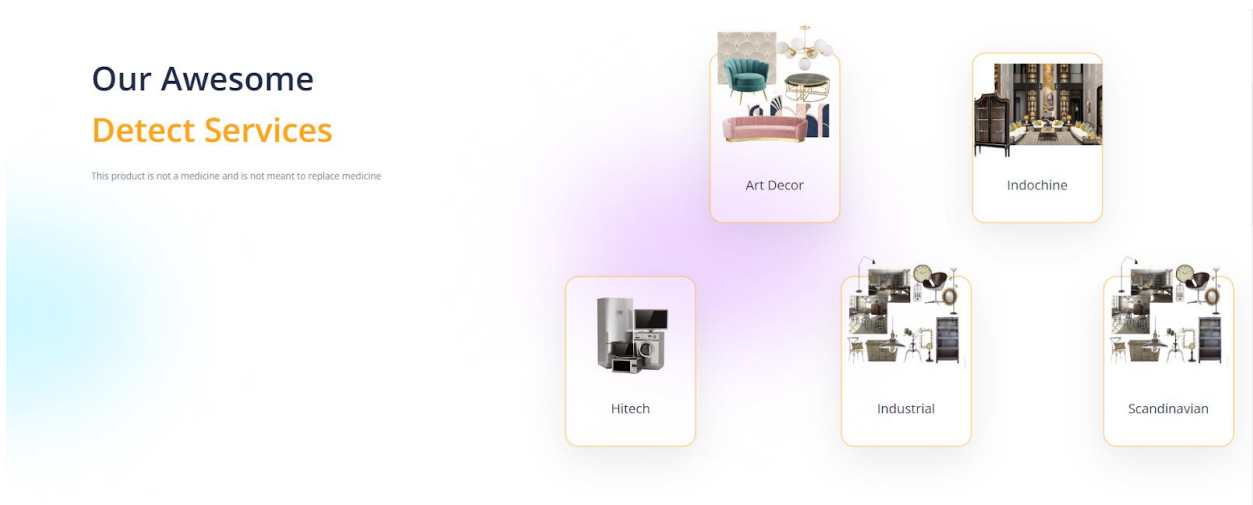


Figure 32: Feature introduction screen

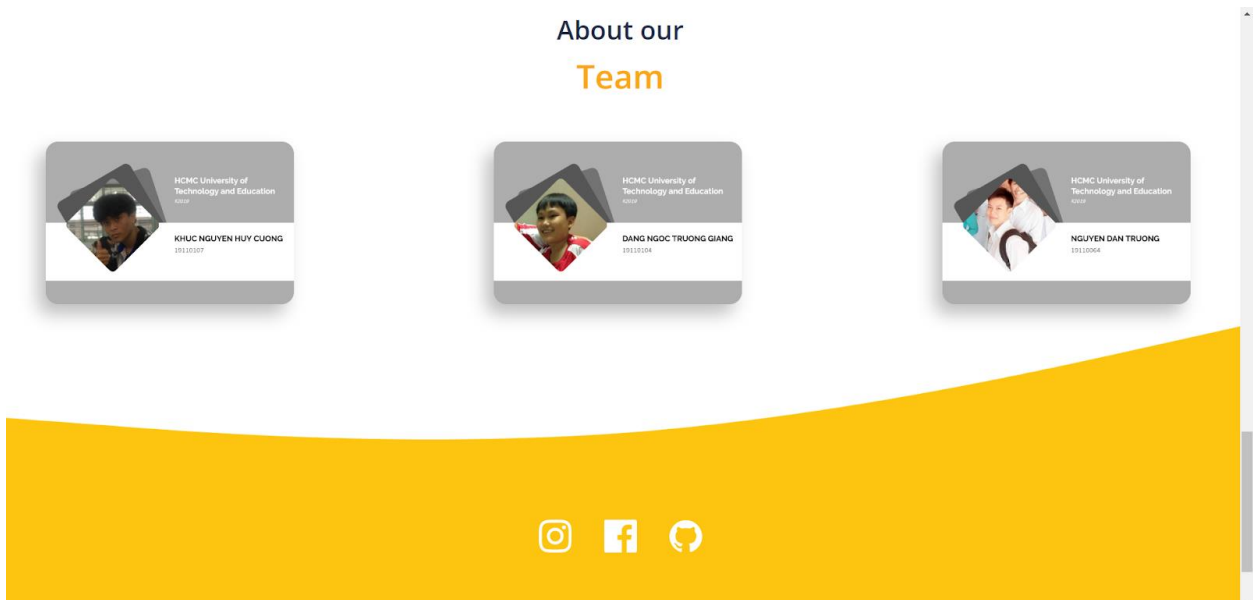


Figure 33: Group member introduction screen

After upload an image file and detect the application will render a list of object which have detected and it ratio to each interior furniture styles

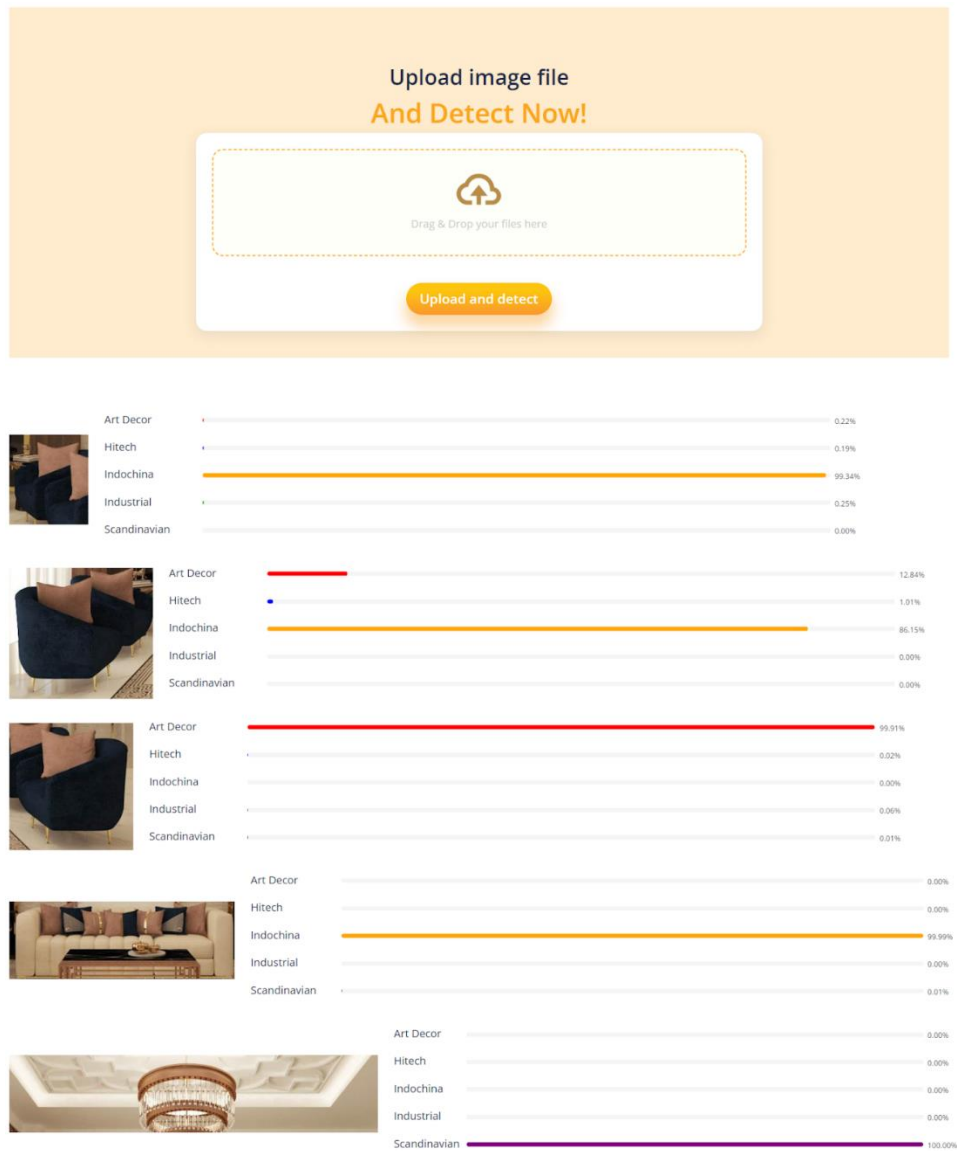


Figure 34: Demonstration for application

In addition, on each object that we have discovered and give a ratio for each interior style, if customers do not agree with us, they can choose the icon dislike and choose the interior styles they think that object was belong to. We will collect their data and verify for dataset become better in future.

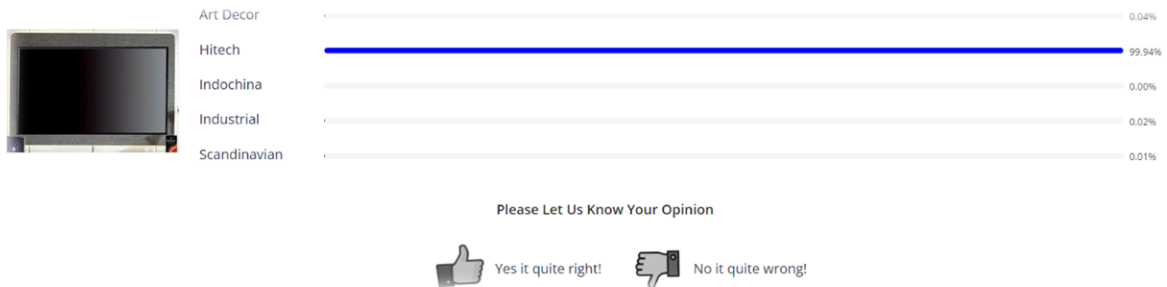


Figure 35: Evaluate crop image

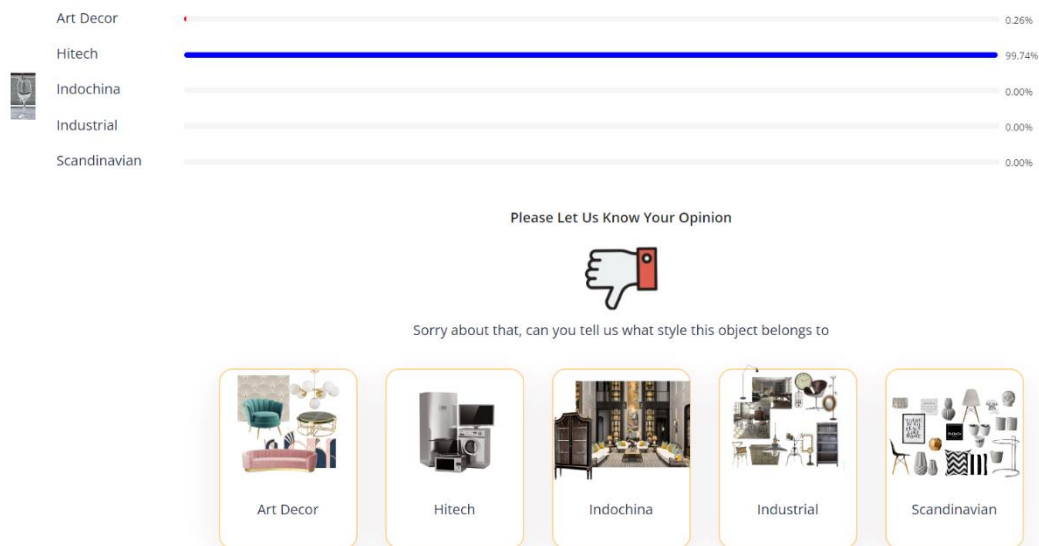
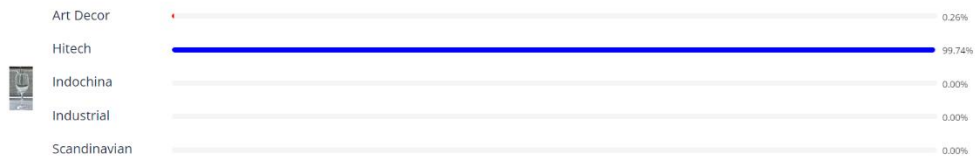


Figure 36: Client evaluate wrong style 1



Please Let Us Know Your Opinion



Sorry about that, can you tell us what style this object belongs to



Thanks for your support

Figure 37: Client evaluate wrong style 2



Please Let Us Know Your Opinion



Thank you about your feedback!

Figure 38: Client evaluate right style

4.4 Experimental results

Investigate the current model, assess it, and compare it to other models.

The user-friendly website's functionality includes the ability to categorize interior design styles using the models with the best accuracy rate:

The smartphone application's ability to categorize interior design trends using the most accurate models is one of its many features.

CHAPTER 5: CONCLUSIONS AND FUTURE WORKS

5.1 Summary

We discovered the true interior design process while working on this project, and we also learnt and understood more about Deep Learning and Machine Learning. We were able to address design style categorization issues in our project thanks to Deep Learning. Building and successfully implementing a design style prediction API using static image, successfully creating a basic web page with built-in API and photo-based predictive design style, and also creating a power app that uses a previously built API to categorize design styles of interior of each object is determined by yolov5.

5.2 Limitations

Some of the project's shortcomings stem primarily from the lack of training materials, most of which are free. Many classes can have significantly higher guess rates than others, especially when there is a large imbalance between classes and insufficient data.

5.3 Future works

Our project's future aim is to learn about the GAN (Generative Adversarial Networks) deep learning technology and design applications that can generate furniture based on training data. We will strive to expand the number of classes and samples in each class as much as feasible in order to enhance the prediction rate more correctly in practice. Provide extra apps that help in interior design in addition to speeding up the process and lowering its expenses. At the same time, we will build more mobile applications in the near future to make it more convenient for consumers.

REFERENCES

1. Sciences, A. I. (2022, February 16). *VGG19 architecture & implementation | Image Classification | Deep learning*. YouTube. https://www.youtube.com/watch?v=V_O9QtVg8bY&feature=youtu.be.
2. Module: *tf.keras.applications* | TensorFlow v2.11.0. (n.d.). TensorFlow. https://www.tensorflow.org/api_docs/python/tf/keras/applications
3. Kaushik, A. (2020, July 21). *Understanding ResNet50 architecture*. OpenGenus IQ: Computing Expertise & Legacy. <https://iq.opengenus.org/resnet50-architecture/>.
4. Narein T, A. (2021, October 8). *Inception V3 Model Architecture*. OpenGenus IQ: Computing Expertise & Legacy. <https://iq.opengenus.org/inception-v3-model-architecture/>.
5. *Xception Model and Depthwise Separable Convolutions*. (2019, March 20). <https://maelfabien.github.io/deeplearning/xception/>
6. Sciences, A. I. (2022c, February 16). *VGG19 architecture & implementation | Image Classification | Deep learning*. YouTube. https://www.youtube.com/watch?v=V_O9QtVg8bY&feature=youtu.be.
7. Develop. Preview. Ship. For the best frontend teams –. (n.d.). Vercel. <https://vercel.com/>.
8. Géron, A. (2022). *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems* (3rd ed.). O'Reilly Media..
9. Ramstedt, F., & Olofsson, M. (2020). *The Interior Design Handbook: Furnish, Decorate, and Style Your Space* (Illustrated). Clarkson Potter..
10. Solawetz, J. (2022, November 21). *What is YOLOv5? A Guide for Beginners*. Roboflow Blog. <https://blog.roboflow.com/yolov5-improvements-and-evaluation/>.
11. Ramstedt, F., & Olofsson, M. (2020). *The Interior Design Handbook: Furnish, Decorate, and Style Your Space* (Illustrated). Clarkson Potter.
12. Chollet, F. (2021). *Deep Learning with Python, Second Edition* (2nd ed.). Manning.
13. Mì AI. (2022, August 31). *Thử build và run docker image cho ứng dụng Deep Learning - Mì AI*. YouTube. <https://www.youtube.com/watch?v=TOCxu0jh8ic>.
14. Mì AI. (2020, November 22). *[Mì Python] Bài 7. Dựng Backend server bằng Flask - Mì AI*. YouTube. <https://www.youtube.com/watch?v=PX-ayulp-kk>.
15. *ReactJS Documentation* –. (n.d.). React - Meta Platforms, Inc. - Facebook Open Source. <https://reactjs.org/docs/getting-started.html>