Bilkent University
Department of Computer Engineering

Senior Design Project

Kalas-Iris: Clothes recognition and rich attribute prediction using computer vision service for online clothing retail

Analysis Report

Olcay Akman, Doğaç Eldenk, Zeynep Korkunç, Hüseyin Ata Atasoy

Supervisor: Ayşegül Dündar

Jury Members: Çiğdem Gündüz Demir and Can Alkan

Analysis Report
November 16, 2020

This report is submitted to the Department of Computer Engineering of Bilkent University in partial fulfillment of the requirements of the Senior Design Project course CS491/2.

Table of Contents

1 Introduction

Online shopping is becoming more and more popular. There are around 7.1 million online retailers in 2020 and 1.8 billion people are shopping online each year. The demand on online shopping is requiring more resources to be spent for customers. There are two main ways customers find the goods that they buy in those websites: Browsing and searching. According to Oberlo [1], 75% of the search queries of the customers are brand new. Therefore, it is important for customer to find the good that they are searching for.

Currently big e-commerce websites such as Hepsiburada use textual matching for searching products. To get better results, most of the products are labeled and categorized. In addition to those labels, categories and descriptions, there are additional handwritten rules for better customer experience. This is very expensive and hard to maintain even for big e-commerce websites. Currently the search results of those products are not inspired from the images of the product. Machine learning is only used for showing relevant products to another product, which is affected by the behavior of the customers.

Most of the current search engines of small e-commerce websites recommend a good match to the given search query via matching the search queries with the labels associated with each product. Although successful to provide the customer with the products they are looking for, such services lack a high rate of accuracy. This mainly stems from the fact that small e-commerce owners do not have the flexibility to spend extensive time and effort on keeping a thorough and detailed log for their inventory on a digital platform. Required information about goods are put in the system by hand and sometimes directly gathered from the distributor. When the information lacks in the system, workers add the description based on the image of the product. Additionally, the products have different filters and labels. Labeling these products are time consuming and there can be missing information. The amount of detail in those descriptions also can be limited. Additionally, there can be semantic problem such as synonym or similar words. In order to overcome those problems, we want to create an algorithm that can create descriptions and labels for items and also can show the best matching products according to a search query. We expect this algorithm to benefit from mostly the images of the products and additionally descriptions.

2 Current System

DeepFashion is a large-scale clothes dataset developed by Multimedia Laborataory of the China University of Hong Kong [2]. It contains over 250,000 images labeled with 50 categories and 1000 attributes. This dataset is used to create the open-source visual fashion-analysis toolbox, MMFashion [3] in 2019. MMFashion is still under development and not a complete product. It can predict attributes, recognize features, detect landmarks, parse, segmentate and recommend clothes. There are papers about attribute and category prediction of fashion clothing. Transfer learning is a commonly used technique for category & attribute prediction of fashion images. VGG-16 [3] and ResNet-50 [3] is used with MMFashion as well. Our goal is to improve on this model and serve it as an automatic tagging service. Additionally, we are going to benefit from those categories and features for better search algorithms.

3 Proposed System

3.1 Overview

Kalas-Iris is going to be a web application providing two services to the user: image category and attribute prediction service, semantic search service. Image category and attribute prediction service is going to get an image from the user, online retail store is our user in this case, and it is going to send back the predicted category and attributes of the image. Category and attribute prediction service can be used with an automatic webhook or manually using a web interface. Semantic search service will be used by the user, customer of the retail store in this case, and it will retrieve the best items matching to the search query.

3.1.1 Image Category and Attribute Prediction

Deep learning algorithms are vastly used for image classification and segmentation problems. Kalas-Iris's category and attribute prediction system is going to benefit from modern deep learning techniques. DeepFashion is a state-of-the-art paper specialized on clothes recognition, retrieval and annotation [4]. It contains more than 800,000 samples and it is a great starting point for our model. The proposed network, FashionNet, is using VGG-16 as a backbone. The last convolutional layer of VGG-16 is replaced by three branches of layers carefully designed for clothes. In MMFashion, ResNet-50 is performing slightly better than VGG-16 with better recall values [3]. Therefore, an alternative to the FashionNet can be MMFashion architecture, backboned with ResNet-50 using a simple head layer. But recent studies shown that Attentive Fashion Grammar Network outperforms FashionNet in almost every category and attribute [5]. Attentive Fashion Grammar Network uses VGG-16 with Bidirectional Convolutional Recurrent Neural Network (BCRNN) [6]. A possible experiment can be done using ResNet-50 with BCRNN to compare the results. Finally, in the paper of Amazon.com, Deep Learning for Automated Tagging of Fashion Images, the problems of perturbations and deformations on images are discussed [7]. It is shown that a training with a smaller architecture such as AlexNet or a regularizing network such as GoogleNet outperformed ResNet-18. Therefore, it is possible to use many different architectures with different drawbacks. We are going to experiment with different architectures to see the results.

3.1.2 Semantic Search

The Semantic Search part of our tool aims to be functional to the customer of the e-commerce website by providing them with accurate search results. Via personal experience, we observed that many current online retail shops provide a poor search service for their products. An important part of getting accurate search results lies in giving proper labels to the search items. This part is achieved by the Fashion Classifier offered by our service. Another part of increasing search result accuracy depends on the semantic search algorithms used. For this purpose, we researched Word2Vec and Doc2Vec algorithms. The Word2Vec algorithm, simply put, converts words to vectors and computes their cosine similarities to train the model. This can be implemented via the Skip-Gram [8] or Common Bag of Words (CBOW) [9] algorithms. We are going to experiment with these models and depending on the results we obtain; we will decide on which method to use. In this regard, it is crucial that our Fashion Classifier works well with our Semantic Search algorithm. The Fashion Classifier will label clothing items in a certain pattern, which will result in a consistent set of labels and categories. This feature of Kalas-Iris will be highly valuable for the Sematic Search part of it, as it will yield more consistent and reliable search results. This way, the data that the Search Algorithm will be trained on will be consistent. On another note, when the Fashion Classifier is actively in use, the incoming data for the Semantic search algorithm will be ever-changing and ever-expanding. In order to help adapt our search algorithm to this flexible data set of clothing labels, we will experiment with some Dynamic Model Averaging (DMA) methods for our model. This is an active approach in use for different areas such as soil moisture estimates [10], forecasting carbon emissions [11], and house price forecasting [12]. DMA is a flexible method and is used in many different areas, thus it will likely be useful for our project.

3.2 Functional Requirements

3.2.1 User Account

- The system should provide its users to create an account in order for them to benefit from its services.
- Customers can add their website's end points to the system for automatic image labeling service.
- Kalas-iris customers should be able to obtain a private key from the web application.
- By using the key, customers should be able to make requests to the system in order for them to get the required results for their images.

3.2.2 Image Category and Attribute Prediction

- Customers can upload single images or use custom webhooks for uploading an image of the new item to the system.
- The system will automatically predict the category and the attributes of the given item.
- The system will send the results to the customer by making a http request to the customer's website or showing a simple webpage.

3.2.3 Automatic Category and Attribute Tagging Service

- The system will predict of the category and attributes of an item using its image.

- The system will benefit from a pre-trained model using modern Deep Learning techniques.
- The system will be available on an endpoint in the webapp.

3.2.4 Semantic Search Service

- Kalas-iris should be able to enhance their customers' search services in their websites by predicting the attributes from the images of the product.

3.3 Non-Functional Requirements

3.3.1 Accuracy

The system should be able to predict the category of the given image 90% of the time. The system should be able to predict at least 3 true attributes of an image. Number of false attributes should not be more than 2.

3.3.2 Availability

The system should be up and running without any disruption. Any disruption to the search service will affect the search performance drastically. Also, image tagging service should warn the company if there is a system problem. The customers should roll back to traditional methods in that case.

3.3.3 Performance

The system should be able to query a single semantic search no more than 2 seconds. The image tagging service should be able to predict the labels in less than 10 seconds.

3.3.4 Integrability

The system should be integrable to existing search API's and inventory management systems. This integration will include a manual image upload page and an automatic web hook for automatically processing the images.

3.3.5 Deployment

The system should be able to deployable to the cloud systems under a minute. This will ensure that the system will have a good uptime. The machine learning models are also required to be deployable on cloud systems. On demand, this system can work in a self-hosted computer.

3.4 Pseudo Requirements

3.4.1 Implementation

The system will be implemented in Python using Django as a backend and PyTorch as a Deep Learning framework. The frontend will be written using React.

3.4.2 Hardware and Software Systems

The system will be trained on a powerful computer with a modern external graphics card for the image tagging service. The final product won't require a high-end computer. The service will be hosted at AWS for the demo.

3.4.3 Legal Concerns

The system will be trained on Deep Fashion dataset. This dataset is not suitable for commercial usage. Our first goal is to see how well this system will work on a tight budget. If we want to improve our system and release it as a final product, we require custom datasets to be made for training the model. This requires too much effort and money.

3.5 System Models

3.5.1 Use Case Model

Scenario 1

Use Case Name:

- Create Account

Actors:

- Online Retail Website Owners

Entry Conditions:

- Moderate an online retail website.

Exit Conditions:

- Account is successfully created.
- Account creation suspended.

Main Flow of Events:

- Owner visits Kalas-Iris website.
- Owner navigates to the sign-up page.
- Owner fills their personal information.
- Owner fills their company information.
- An authentication mail is sent to the company.
- Owner verifies they are working for that company.
- Account is created.

Alternative Flow of Events:

- Owner visits Kalas-Iris website.
- Owner navigates to the sign-up page.
- Owner fills their personal information.

- Owner fills their company information.
- An authentication mail is sent to the company.
- Authentication is failed, account creation is suspended.

Scenario 2

Use Case Name:

- Enter website information

Actors:

- Online Retail Website Owners

Entry Conditions:

- Have a verified account.
- Have the desired endpoints on the webpage.

Exit Conditions:

- Completed and verified the website information.
- Website verification failed.

Main Flow of Events:

- Owner visits Kalas-Iris website.
- Owner navigates to the website information page.
- Owner fills the desired endpoints of their website.
- Owner gets a secret key after the endpoints are verified.

Alternative Flow of Events:

- Owner visits Kalas-Iris website.
- Owner navigates to the website information page.
- Owner fills the desired endpoints of their website.
- Endpoints are not suitable, website verification failed.

Scenario 3

Use Case Name:

- Annotate a single image

Actors:

- Online Retail Website Owners
- People who are curious about the system

Entry Conditions:

- People without accounts can’t make more than 10 requests per day.

Exit Conditions:

- Image is suitable and annotated.
- Image is not suitable.

Main Flow of Events:

- Owner enters Kalas-Iris website.
- Owner signs in to the system.
- Owner navigates to the annotation page.
- Owner uploads an image of a fashion item.
- Owner sees the category and the attributes of the item.

Alternative Flow of Events:

- User enters Kalas-Iris website.
- User navigates to the annotation page.
- User uploads an image of a fashion item.
- User sees the category and the attributes of the item.
- User repeats last two steps 10 times.
- User can’t annotate more images because daily limit is reached.

Scenario 4

Use Case Name:

- Annotate images automatically

Actors:

- Online Retail Website Owners

Entry Conditions:

- Have a verified Kalas-Iris account.
- Owner filled website information correctly.
- Website uses the given secret keys while making the requests.

Exit Conditions:

- Image is suitable and annotated successfully.
- Image is suitable and annotated doubtfully.
- Image is not suitable, annotation aborted.

Main Flow of Events:

- User enters Kalas-Iris website.
- User navigates to image annotation page.
- User uploads a single image to the system.
- The image is categorized and attributes are returned to the user.

Alternative Flow of Events:

- User enters Kalas-Iris website.
- User navigates to image annotation page.
- User uploads a single image to the system.
- The image is unsuitable and the user sees a warning.

Scenario 5

Use Case Name:

- Search items semantically

Actors:

- Online Retail Website Customer

Entry Conditions:

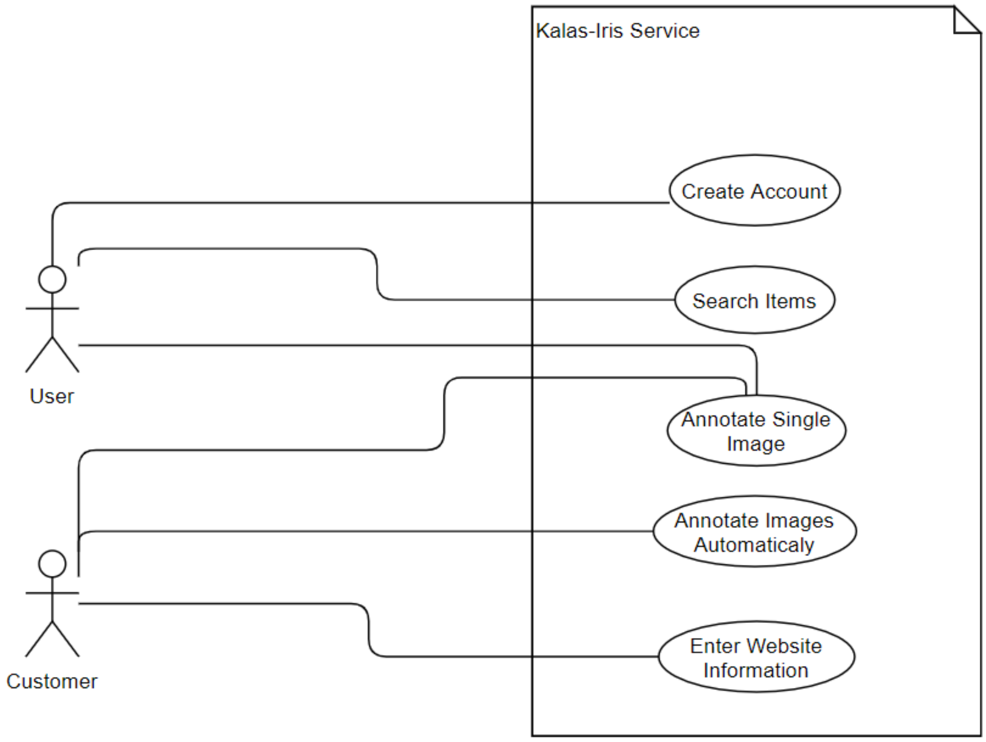
- Website have a verified Kalas-Iris account.
- Website owner filled website information correctly.
- Website implements the Kalas-Iris search API.
- Website uses the given secret keys while making the requests.

Exit Conditions:

- Search results are returned.

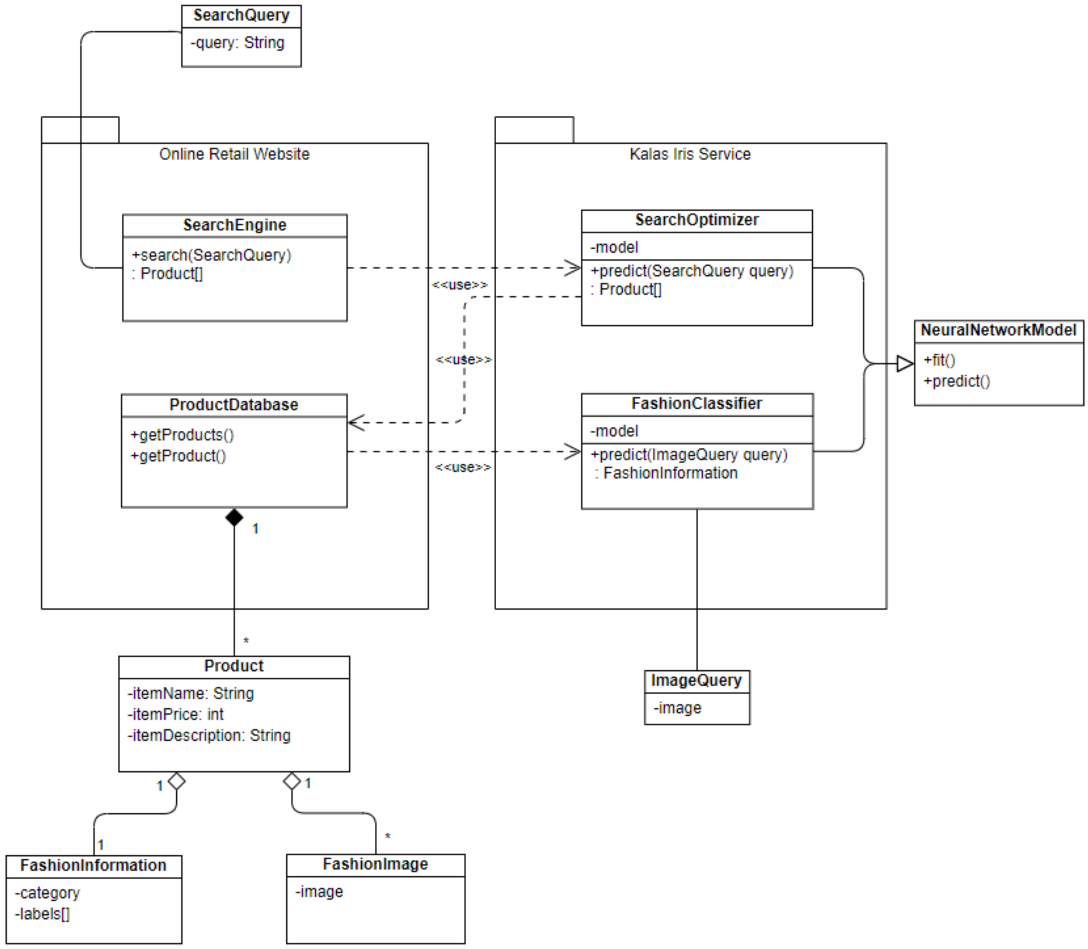
Main Flow of Events:

- User searches an item in the retail store.
- The search query is redirected to Kalas-Iris system.
- Kalas-Iris system ranks the items by the relevance to the query and returns the result.
- User sees the list of items ranked from most relevant to least



Model 1: Use-Case Diagram for Kalas-Iris Website

3.5.2 Object and Class Model



Model 2: Class Diagram for Kalas-Iris Service

SearchQuery: A basic query user generates when searches for some item in the online retail website.

SearchEngine: The engine which evaluates the query and returns the most relevant *Products* from the *ProductDatabase*.

ProductDatabase: A database containing all the items that are sold in the online retail store.

Product: A product that is sold on the online retail store. It contains *FashionInformation* and *FashionImage*. Additionally, it might have price, name and description.

FashionInformation: The Kalas-Iris Service generates FashionInformation automatically from the given FashionImage. Each piece of information has a category and multiple labels.

FashionImage: A single image containing the image of the fashion item. Those images are shot on a studio with a good light.

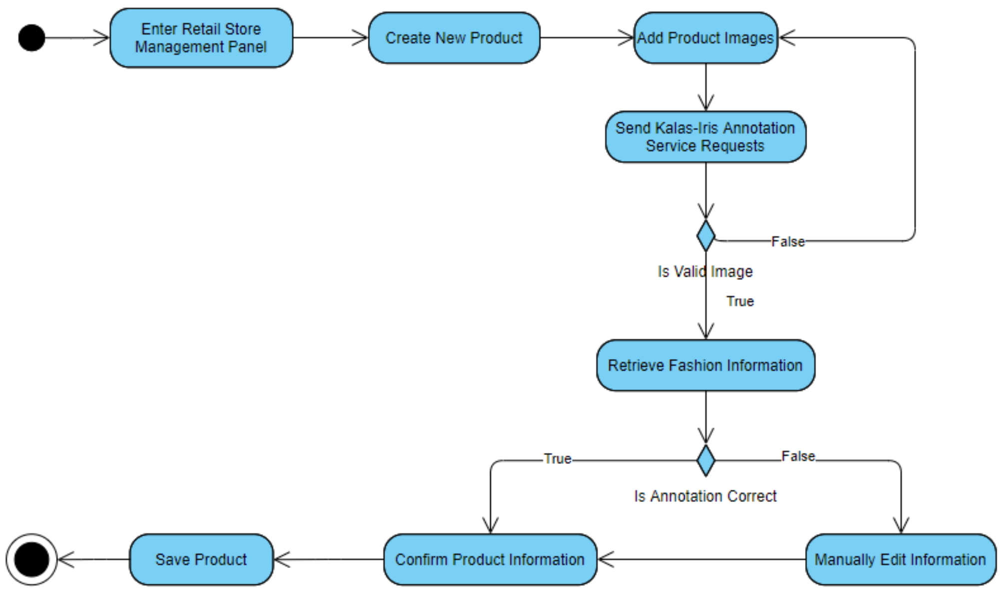
SearchOptimizer: The Kalas-Iris service uses this optimizer to find the most relevant products given a SearchQuery. This optimizer uses NLP techniques and a neural network to predict the products.

FashionClassifier: Generates FashionInformation with the given Image. Provides automated annotation of the clothing item service to Kalas-Iris.

ImageQuery: The image of the product which will be automatically annotated using Kalas-Iris.

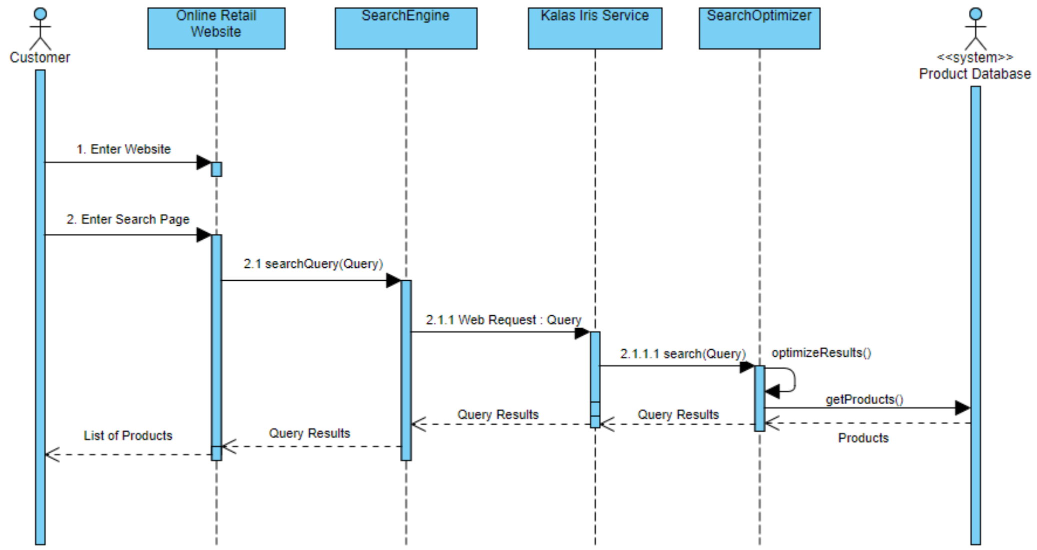
NeuralNetworkModel: A neural network generalization for predictions. Will use PyTorch.NN for this purpose.

3.5.3 Dynamic Models



Model 3: Activity Diagram for Image Annotation

First, the retail store manager enters the management panel of their website. After entering the management panel, they can create a new product. When they add a new image for the product, the website automatically sends a request to the Kalas-Iris annotation service. If the image is not valid, the website asks user to add a new image. After adding a valid image, the Kalas-Iris service suggests categories and labels. The manager can approve that information or can manually edit some parts of it. After editing the information, the product will be added to the system.



Model 4: Sequence Diagram for Searching

In this sequence diagram, a simple search activity is shown. First user enters the website and enters the search page. They will write the search query into the search bar. After clicking search, the website redirects the request to the internal Search Engine. If the Kalas-Iris service is implemented to the engine, it will redirect the request to the Kalas-Iris service. The Search Optimizer will search for the optimal products in the product database. Then the returned products will be redirected cascaded into the website.

3.5.4 User Interface

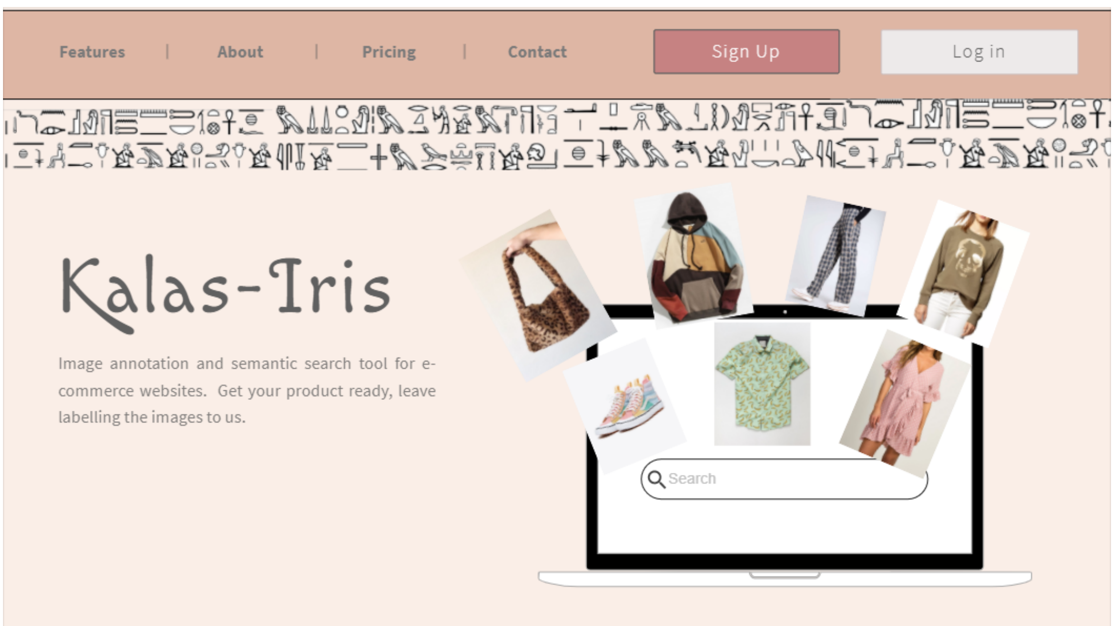


Figure 1: Home Page

This is the home page of the Kalas-Iris service. Most of the functionality can be accessed from here. Also, it will have an appealing look for attracting the customers.

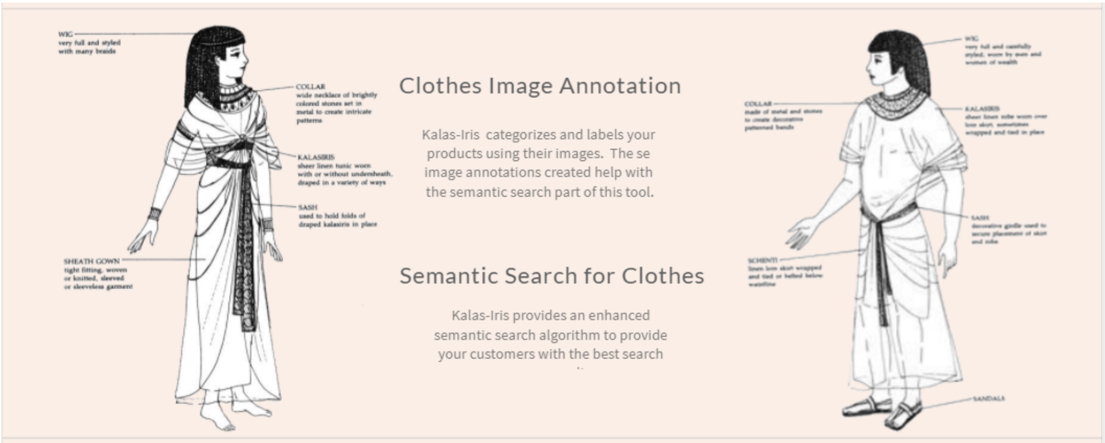


Figure 2: Home Page (Features)

Features page for showing a simple representation for what Kalas-Iris service can do. This shows a brief information to the user about the capabilities of the server.

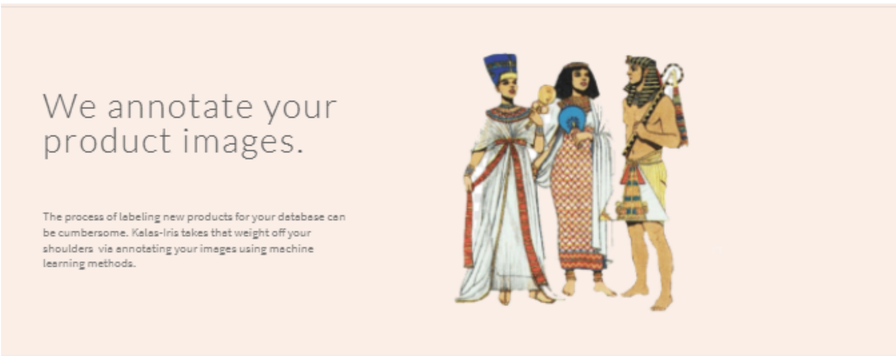


Figure 3: Home Page (About)

A simple about page.

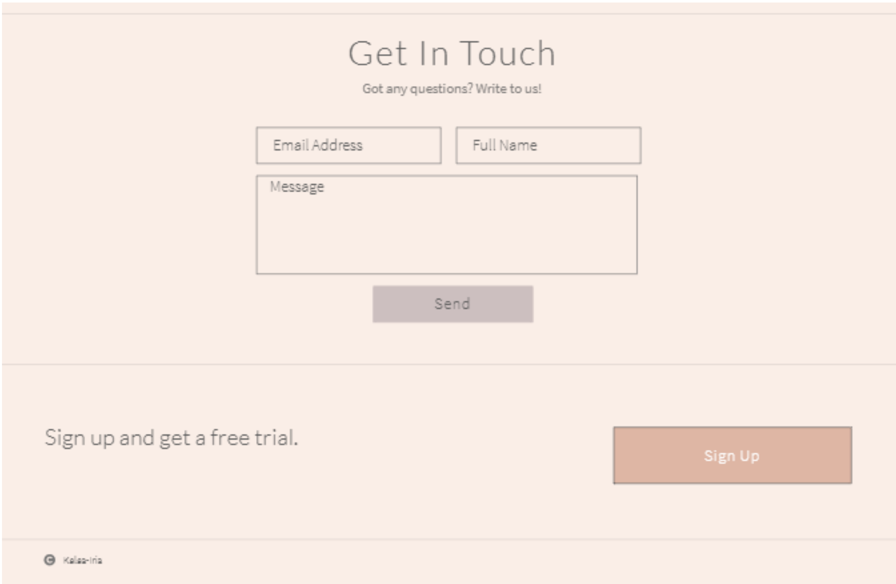


Figure 4: Home Page (Contact)

From this page, customers can get in touch with Kalas-Iris customer services. We will get in touch with the customers personally for creating solutions to the specific needs for their websites. It is important for us to talk personally with our customers.

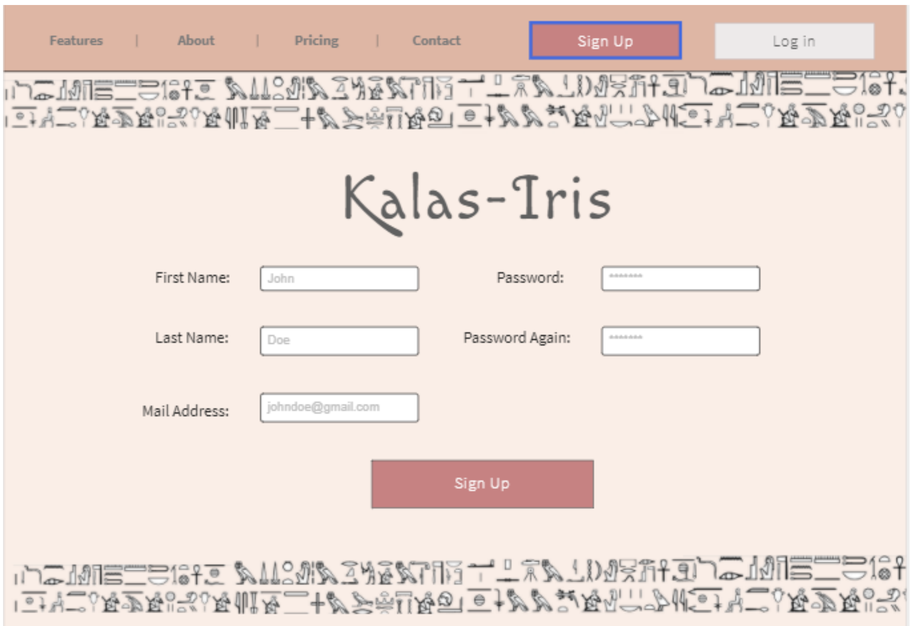


Figure 5: Sign up Page

A very basic sign-up page for Kalas-Iris customers. We do not require much information for signing up. We only require name, last name and mail address.

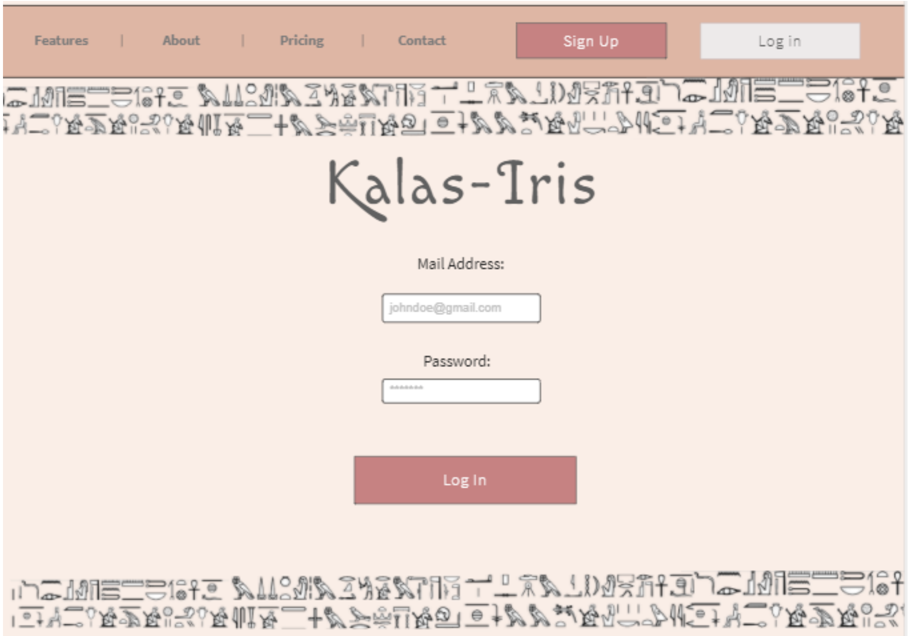


Figure 6: Log in Page

Figure 7: Account Details Page

From this page, customers can retrieve their product key and they can also edit their website URL and endpoints. The Kalas-Iris will send the automated results to the given URL when the annotation is completed. The product key is required by the customer for making requests. The requests are invalid if they do not contain a product key. The customers can also update their website information from this page.

Figure 8: View Products Page

The products page will show the products in the Kalas-Iris database. This is going to be identical to the user's webpage's database. This database is mainly used for exploratory analysis and searching methods.

Figure 9: View Products Page (Labels)

A simple product page.

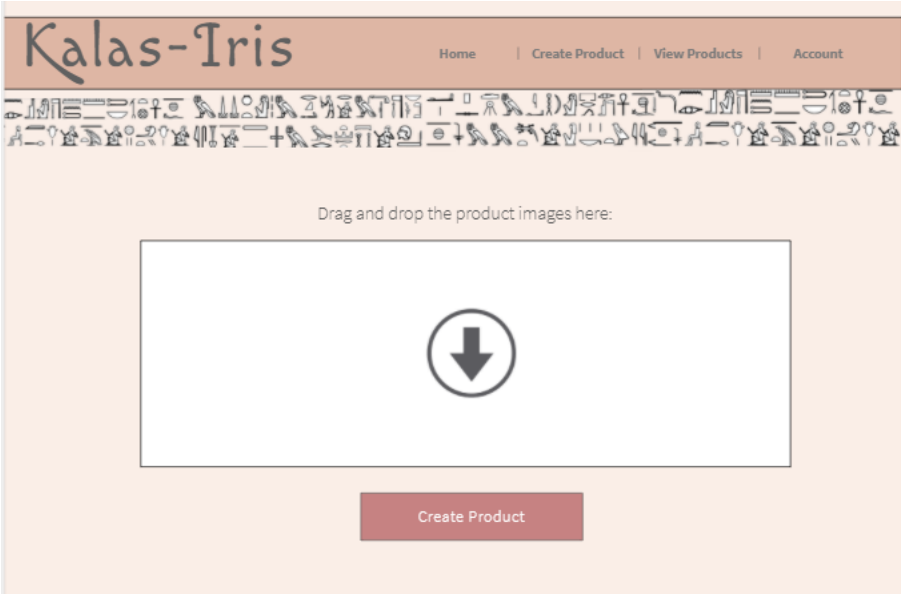


Figure 10: Create Product Page

In this page customers can try with single images to be annotated. This is not the main use case of our product. This will be a demo for the customers. The customer will upload image of a product and the Kalas-Iris service will return the automatically generated FashionInformation to the customer. This information will include category and various labels.

4 Other Analysis Elements

4.1 Consideration of Various Factors in Engineering Design

	Effect Level (Out of 10)	**Effect**
Public Health	1	The ease of search for products in online retailing does not have any immediate effects on public health.

Public Safety	1	Similarly, the ease of search in online retailing does not have any effects on public safety either.
Public Welfare	4	Online retailing is very popular and possible if overall welfare is low.
Global Factors	5	Changes in global fashion industry will affect the accuracy and reliability of our system. Our system should be able to adapt.
Cultural Factors	7	Fashion industry has differences in every culture. The language changes over time so searching algorithms should adapt to the new language.
Social Factors	3	Customers' websites should be protected and not exposed to the other people.

4.2 Risks and Alternatives

	Likelihood	**Effect on the project**	**B Plan Summary**
Wrong Category Prediction	Moderate	Wrongly categorized items will make a hard time for customers to	Human labelers can double check if the confidence of the prediction is low.

		find the items that they have searched for.	
High Costs	Moderate	If the costs are too high, customers might not choose to work with us.	Find partners who are interested in investigating in new technologies rather than small businesses with limited budget.
Outlier Inaccuracies	Low	Outlier items can be placed in a wrong category with wrong attributes.	Manually label outliers using human labelers.

4.3 Project Plan

WP#	**Work Package Title**	**Leader**	**Members Involved**
WP1	Data Finding and Analysis	Hüseyin Ata Atasoy	Olcay Akman Zeynep Korkunç
WP2	Investigating Deep Learning Algorithms	Doğaç Eldenk	Hüseyin Ata Atasoy Olcay Akman Zeynep Korkunç
WP3	Implementing Deep Learning Models	Doğaç Eldenk	Hüseyin Ata Atasoy Olcay Akman
WP4	Testing Deep Learning Models	Olcay Akman	Zeynep Korkunç

WP5	Server Architecture and Design	Olcay Akman	Zeynep Korkunç Doğaç Eldenk
WP6	Frontend Architecture and Design	Zeynep Korkunç	Olcay Akman

WP 1: *Data Finding and Analysis*			
Start date: *8 October 2020* **End date:** 1 January 2021			
Leader:	*Hüseyin Ata Atasoy*	**Members involved:**	*Olcay Akman* *Zeynep Korkunç*
Objectives: Analyze the dataset to calculate bias. Create visual representation for the dataset. Learn analyzing HDF5 data with Python.			
Tasks: ***Task 1.1 Learning h5py for Python*****:** The h5py package is used to analyze data in HDF5 format, which is also the format of Fashion MINST dataset. Read the documentation about the h5py package and follow tutorials on how to use it. ***Task 1.2 Analyze the dataset*****:** Find the amount of data for different categories of fashion items present in the database. Execute the same procedure for each subset of clothes. ***Task 1.2 Report Findings:*** Report the distribution of data based on different types of clothes and create visual representations if necessary.			
Deliverables ***D1.1:*** Report about the distribution of the data.			

WP 2: *Investigating Deep Learning Algorithms*			
Start date: *19 October 2020* **End date:** *14 January 2021*			
Leader:	*Doğaç Eldenk*	**Members involved:**	*Hüseyin Ata Atasoy* *Olcay Akman* *Zeynep Korkunç*
Objectives: Learn what is deep learning. Understand basics of Pytorch. Investigate modern deep learning algorithms that are being used in similar applications.			
Tasks: ***Task 2.1 Learning Fundamentals of Deep Learning:*** Investigate and learn the fundamental idea behind deep learning. 3Blue1Brown YouTube channel has a good video series about explaining deep learning. ***Task 2.2 Learning Basics of PyTorch:*** Read tutorials in PyTorch's website and follow supplementary YouTube videos if necessary. ***Task 2.3 Learn Similar Applications of Deep Learning in Fashion:*** Read papers about fashion recognition, attribute detection and search to understand how to approach to the problem.			
Deliverables ***D2.1:*** A basic application for detecting types of clothes in Fashion MINST dataset.			

WP 3: *Implementing Deep Learning Models for Image Classification*			
Start date: *16 November 2020* **End date:** *1 April 2021*			
Leader:	*Doğaç Eldenk*	**Members involved:**	*Hüseyin Ata Atasoy* *Olcay Akman*
Objectives: Implementing the deep learning models we have figured out. Improving the deep learning models that have been implemented before. Coming up with a final model that can detect attributes and category of a given item.			
Tasks: ***Task 3.1 Implementing Models Defined in Previous Studies:*** Implement the neural network models that are discussed in papers focusing on a similar topic. Test drive the models the see how well they perform. ***Task 3.2 Design and Implement a Model from Scratch:*** Design and implement a model from scratch. ***Task 3.3 Improve and Select a Model:*** Among the implemented models, select the best one and improve it. This will be the final deliverable.			
Deliverables ***D3.1:*** Category and attribute predictor model that will be used in the final demo. It is not going to be fine-tuned. ***D3.2:*** Semantic search model.			

WP 4: *Testing Deep Learning Models*			
Start date: *14 December 2020* **End date:** *1 May 2021*			
Leader:	*Olcay Akman*	**Members involved:**	*Zeynep Korkunç*
Objectives: Testing the implemented Deep Learning Algorithms with test data sets and picking a model.			
Tasks: ***Task 1.1 Preparing the Data Set:*** *Find and prepare the test data set by removing unnecessary data and check it. Find a test data set for both Fashion Classification and Semantic Search.* ***Task 1.2 Applying the Trained Algorithms to Test Data:*** *Use the trained models for Image Annotation and Semantic Search on the prepared data set.* ***Task 1.3 Documenting the Results & Picking a Model:*** *Document the outcome of the testing process and pick the best models by looking at the results.*			
Deliverables ***D1.1:*** *Test Data Set for both Fashion Classification and Semantic Search.* ***D1.2:*** *Documented Results of the tests made for Fashion Classification and Semantic Search.*			

<table>
<tr><td colspan="4">WP 5: Server Architecture and Design</td></tr>
<tr><td colspan="4">Start date: 4 January 2021 End date: 14 April 2021</td></tr>
<tr><td>Leader:</td><td>Olcay Akman</td><td>Members involved:</td><td>Zeynep Korkunç
Doğaç Eldenk</td></tr>
<tr><td colspan="4">Objectives: Designing the server part of the product and implementing the basic server functionalities of Kalas-Iris.</td></tr>
<tr><td colspan="4">Tasks:

Task 1.1 Design the Server Architecture: Plan the server system, its components and its relations with the client side of the service.

Task 1.2 Learn the Necessary Technology for Implementation: Learn Back-End development technologies such database management systems and programming languages.</td></tr>
<tr><td colspan="4">Deliverables

D1.1: The backend service of Kalas-Iris.</td></tr>
</table>

WP 6: *Frontend Architecture and Design*			
Start date: *8 February 2021* **End date:** *1 May 2021*			
Leader:	*Zeynep Korkunç*	**Members involved:**	*Olcay Akman*
Objectives: Creating a user-friendly website by conducting a research about user experience and design a user interface architecture.			
Tasks: ***Task 6.1 Learning React for Front-end Development:*** *Watch tutorials and courses in order to learn the fundamentals of React.* ***Task 6.2 Design the User Interface*** *Conduct appropriate research about both UX and UI designs. Design a user-interface which is both aesthetical and functional for the users.* ***Task 6.3 Design & Implement the Front-End Architecture*****:** *Design a front-end architecture and implement it with the learned technologies.*			
Deliverables ***D1.1:*** *A user-friendly website for Kalas-Iris.*			

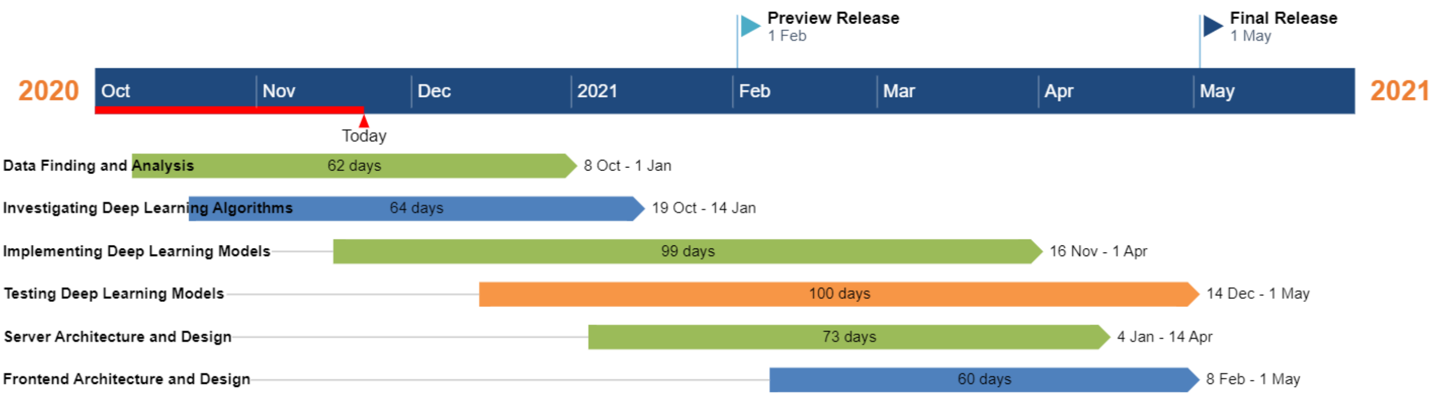


Chart 1: Gantt Chart for Product Development Timeline

4.4 Ensuring Proper Teamwork

We will be using GitHub for collaborating over the code. We will be using GitHub issues for the project management. WP3, WP4, WP5, and WP6 will have their own branches. We will regularly review each other's code and we will use merge requests to keep everything clean. We will also use VSCode's LiveShare feature for pair programming remotely. We also have a discord channel for communication. We are also keeping the important links in the channel. We are using Office 365 for writing the reports together. We have a common OneDrive folder for sharing files. We also opted to use Basecamp, which allows us to schedule meetings, set-up "To-Do" lists, and decide on deadlines, all on a single platform.

4.5 Ethics and Professional Responsibilities

Our project aims to provide the best possible experience in online clothing retail for both customers and e-commerce websites. For the customers, we must ensure the privacy of their information and fit to the KVKK [13] and GDPR [14] regulations of Turkey and EU, respectively. Another point we have to consider is to make sure our semantic search algorithm provides unbiased results, regarding products from competitor brands.

4.6 Planning for New Knowledge and Learning Strategies

Since there is not a member in the team, who is familiar with Deep Learning or Neural Networks, we plan on using online lectures and tutorials which are about these concepts.

Other than online tutorials or lectures, we have been reading papers related to our project in order to familiarize ourselves with clothes recognition and visual fashion analysis topics.

5 Glossary

Kalas-Iris: The name of our service. Kalasiris is a distinctive and important garment worn by women in the ancient Egypt. The Iris is a verbal trick referring to the eye which classifies the images of the clothing items.

Fashion Image: An image of any clothing item. This is not a customer image; it is shot in a studio with good lightning conditions.

Fashion Information: Information of a clothing item that contains category and various labels. Those labels will be striped, long sleeve, tuck tie etc.

Customer: Depending on the scenario, the customer can be the customer of the online retail store or Kalas-Iris web service.

Image Annotation: Annotating an image with a list of words, labels or categories that it belongs to. This process is automated by the neural networks.

Semantic Search: Searching a list of products with the use of the natural language we use every day. This includes understanding synonyms and contextual text.

6 Works Cited

[1] M. Moshin, "10 ONLINE SHOPPING STATISTICS YOU NEED TO KNOW IN 2020," Oberlo, 23 March 2020. [Online]. Available: https://www.oberlo.com/blog/online-shopping-statistics.

[2] Z. Liu, P. Luo, S. Qiu, X. Wang and X. Tang, "DeepFashion Database," June 2016. [Online]. Available: http://mmlab.ie.cuhk.edu.hk/projects/DeepFashion.html.

[3] X. Liu, J. Li, J. Wang and Z. Liu, "MMFashion: An Open-Source Toolbox for Visual Fashion Analysis," Hong Kong, 2020.

[4] Z. Liu, P. Luo, S. Qiu, X. Wang and X. Tang, "DeepFashion: Powering Robust Clothes Recognition and Retrieval with Rich Annotations," CVPR, Hong Kong, 2016.

[5] W. Wang, Y. Xu, J. Shen and S.-C. Zhu, "Attentive Fashion Grammar Network for Fashion Landmark Detection and Clothing Category Classification," CVPR, Los Angeles, 2018.

[6] Q. Dong, S. Gong and X. Zhu, "Multi-task Curriculum Transfer Deep Learning of Clothing Attributes," IEEE, Santa Rosa, CA, 2017.

[7] P. Gutierrez, P.-A. Sondag , P. Butkovic, M. Lacy, J. Berges, F. Bertand and A. Knudson, "Deep Learning for Automated Tagging of Fashion Images," CVPR, 2018.

[8] T. Mikolov, I. Sutskever, K. Chen, G. Corrado and J. Dean, "Distributed Representations of Words and Phrases and their Compositionality," Google Inc., 2013.

[9] D. Karani, "Introduction to Word Embedding and Word2Vec | by Dhruvil Karani | Towards Data Science," 1 Sep 2018. [Online]. Available: https://towardsdatascience.com/introduction-to-word-embedding-and-word2vec-652d0c2060fa.

[10] Y. Huiling, Y. Yize, Y. Yize and S. Ruochen, "Sub-daily soil moisture estimate using dynamic Bayesian model averaging," *Journal of Hydrology,* no. 590, 2020.

[11] S. Xu, Y. Zhang and X. Chen, "Forecasting Carbon Emissions with Dynamic Model Averaging Approach: Time-Varying Evidence from China," *Discrete Dynamics in Nature and Society,* vol. 2020, 2020.

[12] Y. Alisa, N. G. Pavlidis and E. G. Pavlidis, "Adaptive Dynamic Model Averaging with an Application to House Price Forecasting," 2019.

[13] "Mevzuat Bilgi Sistemi," 24 3 2016. [Online]. Available: https://www.mevzuat.gov.tr/mevzuat?MevzuatNo=6698&MevzuatTur=1&MevzuatTertip=5.

[14] "General Data Protection Regulation - (GDPR) Official Legal Text," 25 May 2018. [Online]. Available: https://gdpr-info.eu.

[15] C. Corbiere, H. Ben-Younes, A. Rame and C. Ollion, "Leveraging Weakly Annotated Data for Fashion Image Retrieval and Label Prediction," ICCV, Paris, France, 2017.

[16] W. Di, C. Wah, A. Bhardwaj, R. Piramuthu and N. Sundaresan, "Style Finder: Fine-Grained Clothing Style Detection and Retrieval," IEEE, Portland, OR, 2013.