A History of Fashion through the Lens of Machine Learning

Irina Hallinan[1,✉]

[1]CS 294-26 Intro to Computer Vision and Computational Photography, UC Berkeley

Fashion is a reflection of society. The clothes we wear communicate non-verbal information, such as a person's social status, affiliation, class, and mood. Clothes are powerful symbols that mirror societal trends. In this project, fashion photographs are analyzed through the lens of historical events. First, three deep neural networks are trained and tested to extract information about fashion photographs and then applied to photographs from American fashion magazines. Second, summaries of historical context are extracted from articles for each decade starting in the 1970's until the 2020's, using natural language processing. Third, a morph sequence is created for each fashion photograph in a time series. Afterwards, the morph sequence together with historical context are combined into a timeline video. The video shows changes in fashion photographs throughout each decade alongside historical context. This work aims answer questions like what garment is the most popular in fashion magazines at a particular decade, and what political events occur during which fashion trends seen in the photographs.

vision | neural networks | fashion | history | natural language processing
Correspondence: ***irina_hallinan@berkeley.edu***

Introduction

From the dawn of human history, people were interested in adoring their bodies in clothing and jewelry (1). Fashion as its own industry evolved in Europe in the 19th century (2). By the beginning of the 20th century, there were fashion magazines, such as Vogue, where readers could get a first look at what garments to wear. Presently, clothing, high fashion, and apparel industries are huge parts of many global economies, with trillions of dollars in yearly revenue (3). The recent advanced in Artificial Intelligence (AI) and deep learning in particular impact both retailers and consumers, as well as clothing advertisers and manufacturers. In addition to fashion businesses, classification of clothing photographs is a topic of interest to a wide range of people, including computer vision scientists, designers and marketers of clothing (4). There are multiple tasks involved when analyzing clothes: from clothing classification (5) to attribute retrieval (6) to suggestion or similarity engines (7) to more recently, landmark detection (8). The focus of this work is on classification, attribute retrieval, and landmark detection.

Related Work

Recent advances in both hardware and software enable machine learning classification of large data sets of visual data. The authors of the DeepFashion project collect 800,000 publicly-available photographs, including both professional photographs and consumer-posted photos of purchased items, along with their labels (8). The authors create a new comprehensive data set of fashion photographs and conclude that learning multiple categories that may not seem related at first (for example, describing both texture and color of an item) is beneficial to the overall model performance. Therefore, the authors use mixed clothing labels related to multiple aspects of a garment. Several other studies use neural networks to perform garment classification task, such as Brian Lao et al. (9). The authors perform four tasks for classifying clothes: 1) detection, 2) garment classification, 3) attribute classification, and 4) prediction of similar garments. For the clothing type classification problem, the authors use a modified CaffeNet model and the augmented ACS fashion data set. The authors of (10) combine machine learning techniques to analyze both visual photographs and textual historical content of the New York Times archive, in order to see large-scale trends of how cultural events may influence what people wear. A more recent approach by Yeong-Hwa Chang et al. (11) trains a YOLOv5 neural network to recognize clothing styles from images, using the DeepFashion and FashionMNIST data sets. Shenhan Qian et al. (12) integrate the two tasks of classifying and extracting fashion garment key-points using a single network.

Methods

Three deep neural networks are trained to classify clothing garments. Two networks are based on the `ResNet34` architecture and one network is based on the `ResNet18` architecture. The first network is trained to classify fashion garments. The second network is trained to produce multiple clothing attributes of a garment. The third network is trained to produce clothing key-points.

The DeepFashion data set is used for training all three models. The data set is a subset of a larger DeepFashion data set, used for clothing category and attribute prediction tasks. The data set is collected and published by the Multimedia Lab at the Chinese University of Hong Kong. The set contains 289,222 images with bounding boxes and labels, containing 50 clothing categories, and 26 clothing attributes. Version 1.1 of the data set is used, released on December 22, 2016 (8).

A. Neural Networks.

A.1. Clothing Categories. The data set contains 50 distinct clothing categories. The task of predicting clothing cate-

gories is a 1-of-K classification problem. The labels are given as part of the DeepFashion data set annotations. There are 48 unique categories (numbered 1 to 48) because category 49 (Shirtdress) and category 50 (Sundress) have been merged with category 43 (Dress). There are 209,222 training images, 40,000 validation images, and 40,000 test images. Sample data batch for the Clothing Categories model (with text labels and numbers) is shown in the Figures 1a.

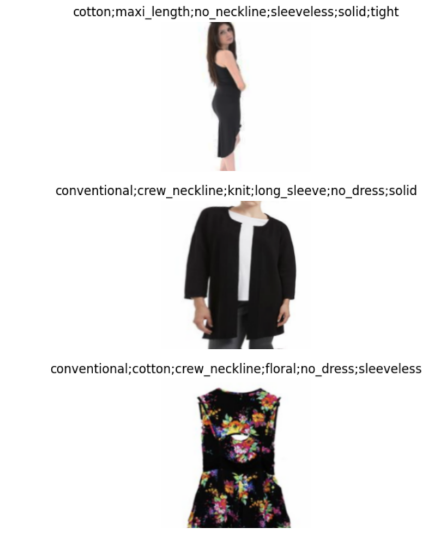
A.2. Clothing Attributes. The data set contains 26 distinct clothing attributes. The task of predicting clothing attributes is a multi-label tagging problem. Each image contains 26 flags, where 1 means a positive label and 0 means a negative label. There are 14,000 images training images, 2000 validation data, and 4000 test images. Sample data batch for the Clothing Attributes is shown in the Figure 1b.

A.3. Clothing Landmarks. The data set contains 8 unique key-points: left collar, right collar, left sleeve, right sleeve, left waistline, right waistline, left hem, and right hem. Missing or occluded key-points are marked with 0's. There are 14,000 training images, 2000 validation images and 4000 test images, along with their bounding boxes and key-points.
For the Landmarks model, the images are converted to gray-scale first. Sample data batch for the Clothing Landmarks model is shown in the Figure 1c.

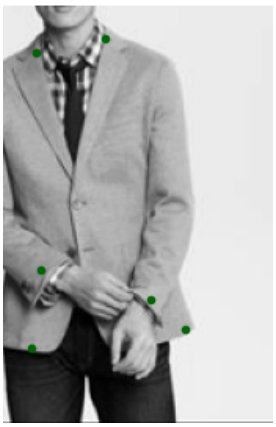
A.4. Data Loader. The data is loaded and augmented using `FastAI` Python library. Each image in the data loader is first resized to be square (300x300 pixels). Then, each image is modified by a random set of augmentations, including rotation, resize, crop, and zoom.
As evident from the images and labels in Figure 1b, not all labels are correct. For example, the dress in the bottom of the image has a label `"no_dress"`. This means that the trained model would predict attributes with a margin of error.

A.5. Training Process. The Clothing Categories model is based on the `ResNet34` architecture with pre-trained weights. After loading the data, the `lr_find` method is used to find the optimal learning rate. The learning rate plotted against the loss curve is shown in the Figure 2a.
For the category model, a `Cross-Entropy Loss (Flattened Loss)` function is used. Knowing the optimal learning rate (around 0.0036308), the model was fine-tuned for 2 epochs. Then, a variable learning rate between 1e-7 and 1e-2 is used to train the tuned model for 6 epochs. The training and validation loss of the model training are shown in the Figure 2b.
Similarly to the Clothing Category model, the model for Clothing Attributes is based on the `ResNet34` architecture and pre-trained weights. The data is loaded and augmented in the same way. The only difference between the attributes model and the category one is the loss function. For Clothing Attributes model, a smoothed version of the default loss function for multi-label tagging problem, `BCEWithLogitsLossFlat` is used. The training and validation loss of the attributes model are shown in the Figure 2c. The model is trained for 10 epochs.

(a) The Clothing Categories Model with category number on the top and label at the bottom.



(b) The Clothing Attributes Model with attribute labels on top of each image.



(c) The Clothing Landmarks Model with key-points marked with green circles.

Fig. 1. Sample data batches

Training either model for longer makes the training and validation losses diverge, which is an indication that the model is over-fitting. The point at which the loss rates start to diverge is around 6 and 10 epochs for categories and attributes model, respectively.

The Landmarks neural network is based on the existing `ResNet18` architecture and pre-trained weights. The model is trained for 100 epochs with a learning rate of 0.0001. For data augmentation, a random color jitter and a random rotation between -5.0 and 5.0 degrees are applied. The `MSELoss` function is used for the problem of predicting key-points. The training and validation loss are shown in the Figure 2d.

A.6. Model Evaluation. The Categories Model achieves around 69.5% accuracy. A sample test batch is shown in the Figure 3a. The Attributes Model achieved accuracy around 87.0%. For the Landmarks Model, successful predictions are shown below, where most key-points are close to the clothing point of interest. Successful predictions are shown in the Figure 3b. Unsuccessful example predictions are shown in the Figure 3c.

A.7. Discussion and Limitations. Both attributes and category models achieve a high accuracy on the test set (above 69%).

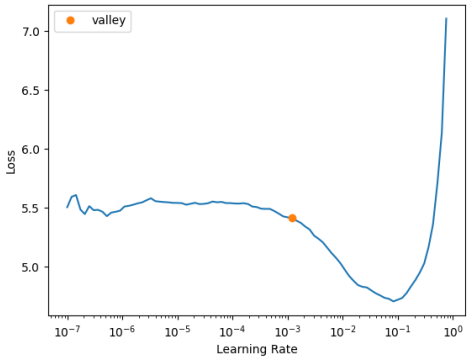
Testing the landmarks network produced mixed results. The predictions seem to do poorly on photographs of models facing away from the camera, and on photos where clothing has a white or light color, perhaps because the color of the clothes is similar to the background.

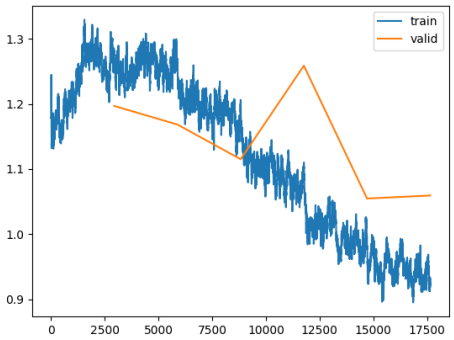
B. Fashion Photographs Data. Fashion photographs are acquired online via web scraping. The web scraper programmatically downloads all of Vogue America covers from https://archive.vogue.com. The time period covered is from the start of the Vogue magazine in 1892 until 2022. Every year has between 10 and 30 magazine covers for a total of 131 years, for a total of 2875 images.

Additionally, a second web scraper downloads images from the Fashion History Timeline website (https://fashionhistory.fitnyc.edu) from 1900 until 2019. Each decade has between 25 and 45 images, for a total of 445 images. The test data set contains a 3320 images between the year 1892 and 2022. Sample Vogue magazine covers and Fashion History Timeline photos are shown in the Figure 4.

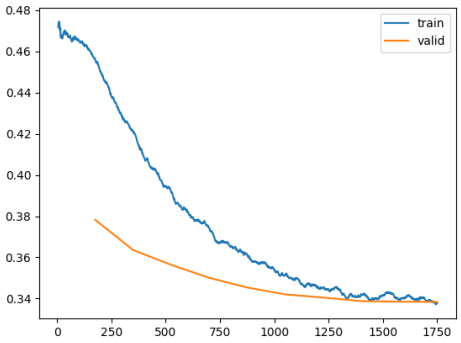
The test data set is prepared by converting each image file path into a line in a Comma-Separated Values (CSV) file. Running the Clothing Categories model on the prepared fashion data set produces a CSV file with predicted categories. Running the Clothing Attributes model on the prepared fashion data set produces a CSV file with predicted categories. Running the Clothing Landmarks model on the prepared fashion data set produces a CSV file with file paths, bounding boxes, and predicted key-points coordinates.

Prior to running the test data set through the trained landmarks model, the Python `cvlib` library is used to automatically detect people and extract bounding boxes around them

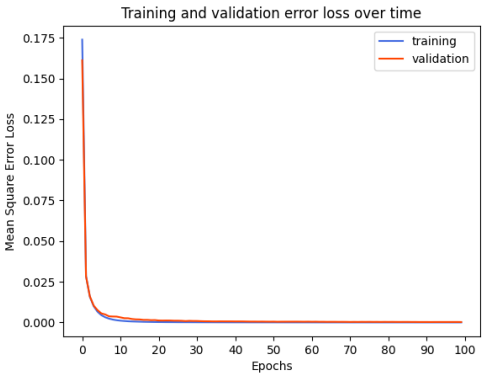




(b) he Categories model training and validation loss curve over time

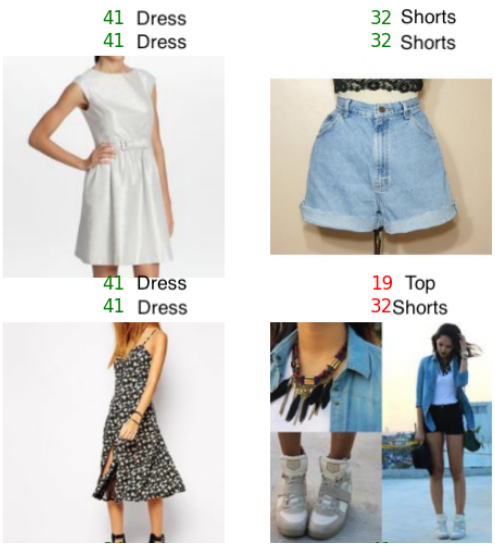


(c) The Attributes model training and validation loss curve over time



(d) The Landmarks model training and validation loss curve over time (seconds).

Fig. 2. Model Training Process



(a) The Categories model evaluation test.

(b) The Landmarks model successful evaluation.

(c) The Landmarks model unsuccessful evaluation.

Fig. 3. Model Evaluation Results

Fig. 4. Sample fashion photographs data set.

in each test photograph. This library uses `YOLOv3` model trained on the COCO data set (https://cocodataset.org/). Only if the detected object has a label "person", the bounding box is extracted and passed to the clothing landmarks model.

C. Historical Events. A list of historical events from the same decade as the fashion photographs is obtained via web scraping and Natural Language Processing techniques. First, a list of relevant articles on https://www.wikipedia.org/ is compiled, describing each decade, starting from the 1970's until the 2020's. For each article, the main content is extracted, excluding the footer, the references, and the side navigation menus. Then, a trained Python `NLTK` package and data are used to produce a summary of the article's content. Each word is weighted by its frequency of occurrence. Then, each sentence is ranked by the frequency of occurring words. The automatic "summary" of the web page is obtained by composing sentences which has the most frequently used words. Top 5 sentences are kept for each summary. The extracted summaries are the historical text that accompanies the fashion photograph morph in the resulting video.

D. Morphing Photographs. Morphing the photograph is done using the Affine transformations after dividing each photograph into a triangular mesh. Details of the morphing technique is described in (13). The correspondences are defined by the 8 landmarks returned by the landmarks neural network model, along with 30 points on the bounding box of the person, and 4 corner points of the image. An example of the 8 predicted key-points is shown in the Figure 5.

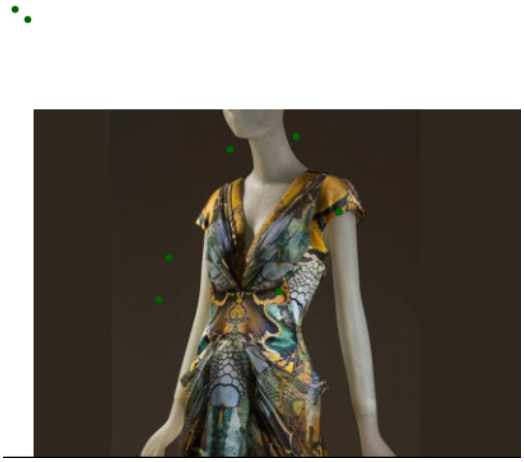


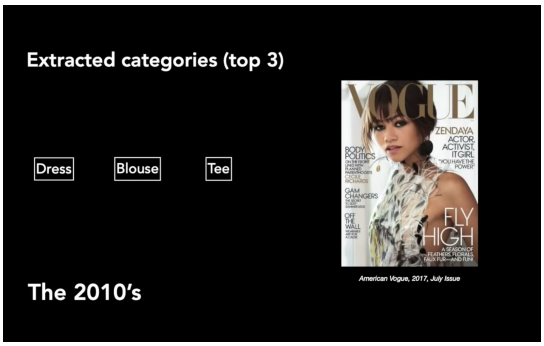
Fig. 5. Landmarks predictions on top of a sample fashion photograph.

Out of 3320 test photographs collected, the photographs from

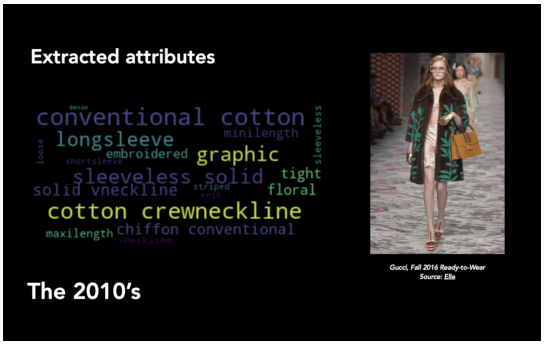
the end of the 19th century and the first half of the 20th century are visually different in appearance from the training data. Therefore, the three models don't return valid results for most of the photographs. Test photographs from the second half of the 20th century until the present pass a visible test. The cutoff for the beginning is 1970. From 1970 until 2022, there are 532 photographs left of the original data set. When discounting invalid bounding boxes and photographs where key-points landed outside the photograph, there are around 300 photographs.
The morph between two photographs consisted of the following steps:

1. Resize each photograph to be the same height (500px).
2. Crop the wider photograph such that the widths of two photographs are the same and all key-points fall inside the photograph.
3. Create a triangular Delaunay mesh of the average (mid) photograph key-points.
4. For each triangle in the mesh, calculate the reverse affine transform to get from source to target triangle.
5. Warp each triangle and the pixels inside it to get the new shape.
6. Shade each triangle with linear interpolation.
7. Repeat for each frame where frame 1 is the source image and frame 45 is the target image, blending both shape and color in each frame proportionally.

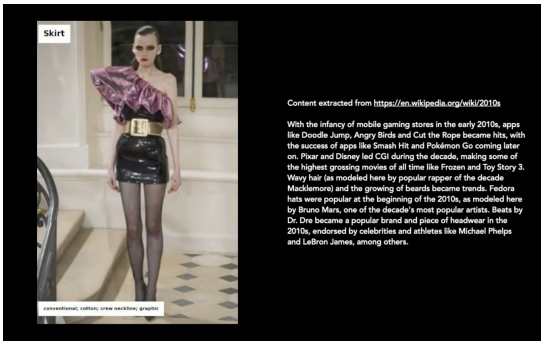
If any step above fails, the photo is discarded from the morph sequence. After running the morphing algorithms on all valid test photos, there are about 100 photographs, which produce a successful morphing sequence.



(a) Frame from the timeline video (attributes).



(b) Frame from the timeline video (attributes).



(c) Frame from the timeline video (morph).

Fig. 6. Frames of the morph video

Results

The morphing video is created after the test data set is evaluated on three neural networks and ran through the morph algorithm. A sample output is shown in the Figures 6a-6c.
The collected fashion photograph data set varies significantly from the modern-day photographs in the DeepFashion data set. Another limitation of the training data set is the 26 categories, which are not inclusive of more nuanced photographs or less commonplace clothing items. Nevertheless, the models accurately predict the category of the main item and get some of the categories in the photograph right, if the photo is similar enough to the training set. In general, the models perform well on later-day photographs and don't perform well on photographs from older magazines (issues prior to 1970) because they are either illustrated by hand or scanned from physical copies. The training set contains only new photographs (post 2000) of digital photographs only.
The outputs of the neural networks are added to each image. Then, each pair of images in a time series for each decade is morphed in 45 frames. The morph transition is sped up to 3 seconds per image pair programmatically, and all the morph sequences together produce the half of the morphing timeline video. Adding the historical context data as the other half and analysis for each decade makes up the rest of the video. The morph video halves are combined the iMovie program. The produced final 3-minute video shows the fashion changes in Vogue and other popular magazines from 1970 until 2022 alongside historical summary of each decade and extracted fashion categories and labels. For each decade, the top 3 extracted categories are displayed before the morph sequence. Attributes are displayed as a word cloud with word size proportional to how frequent that attribute is in the photographs for that decade. Each image in the morph sequence contains the main item category at the top and multiple extracted attributes at the bottom. At a halfway point during the morph sequence, the category label and attributes for the photograph coming into focus replace the starting photograph labels and categories. The full video can be viewed on YouTube. Additional details about this work can be viewed on the project website.

References

1. Carol Andrews. *Amulets of ancient Egypt*. British Museum Press, 1994.
2. History of design, https://en.wikipedia.org/wiki/history_of_fashion_design, 2022.
3. Carolyn Maloney. The economic impact of the fashion industry. *US House of Representatives*, 2015.
4. Yannis Kalantidis, Lyndon Kennedy, and Li-Jia Li. Getting the look: clothing recognition and segmentation for automatic product suggestions in everyday photos. In *Proceedings of the 3rd ACM conference on International conference on multimedia retrieval*, pages 105–112, 2013.
5. Lukas Bossard, Matthias Dantone, Christian Leistner, Christian Wengert, Till Quack, and Luc Van Gool. Apparel classification with style. In *Asian conference on computer vision*, pages 321–335. Springer, 2012.
6. Huizhong Chen, Andrew Gallagher, and Bernd Girod. Describing clothing by semantic attributes. In *European conference on computer vision*, pages 609–623. Springer, 2012.
7. Si Liu, Zheng Song, Meng Wang, Changsheng Xu, Hanqing Lu, and Shuicheng Yan. Street-to-shop: Cross-scenario clothing retrieval via parts alignment and auxiliary set. In *Proceedings of the 20th ACM international conference on Multimedia*, pages 1335–1336, 2012.
8. Ziwei Liu, Ping Luo, Shi Qiu, Xiaogang Wang, and Xiaoou Tang. Deepfashion: Powering robust clothes recognition and retrieval with rich annotations. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.
9. Brian Lao and Karthik A. Jagadeesh. Convolutional neural networks for fashion classification and object detection. 2015.
10. Wei-Lin Hsiao and Kristen Grauman. From culture to clothing: Discovering the world events behind a century of fashion images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1066–1075, 2021.
11. Yeong-Hwa Chang and Ya-Ying Zhang. Deep learning for clothing style recognition using yolov5. *Micromachines*, 13(10):1678, 2022.
12. Shenhan Qian, Dongze Lian, Binqiang Zhao, Tong Liu, Bohui Zhu, Hai Li, and Shenghua Gao. Kgdet: Keypoint-guided fashion detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 2449–2457, 2021.
13. Morphing faces, https://inst.eecs.berkeley.edu/ cs194-26/fa22/upload/files/proj3/cs194-26-ady/, 2022.