IMAGE CLASSIFICATION

Submitted by:

SHILPA KEWALARAMANI

ACKNOWLEDGMENT

I would like to acknowledge the contributions of the following people without whose help and guidance this report would not have been completed. I acknowledge the counsel and support of our training coordinator, **Mr. Shubham Yadav**, SME of Internship14 and **Mr. Sajid Chaudhary SME** of Internship15 with respect and gratitude, whose expertise, guidance, support, encouragement, and enthusiasm has made this report possible. Their feedback vastly improved the quality of this report and provided an enthralling experience.

I am indeed proud and fortunate to be supported by them. I am also thankful to **Miss. Vaishali Singh** HR **of FlipRobo Technologies**, and **Miss Neha** Mentor of Batch No 1827 of **DataTrained Institute, Noida** for her constant encouragement, valuable suggestions and moral support.

Although it is not possible to name individually, I shall ever remain indebted to the faculty members **of Data Trained Institute of Technology & Management, Noida for** their persistent support extended during this work.

This acknowledgement will remain incomplete if I fail to express our deep sense of obligation to my parents and God for their consistent blessings and encouragement

INTRODUCTION

- **Business Problem Framing**

  Describe the business problem and how this problem can be related to the real world.

  Earlier the word ‘data’ was used for representing some text information or some files containing details of some items and the data comprises of everything was in the format of ‘text’ . But today with the tremendous use of smart devices the data is generating at a speed of TB and the most important thing is today data not only refers to text but also to images.

  Image Classification is a fundamental task that attempts to comprehend an entire image as a whole. The goal is to classify the image by assigning it to a specific label. Typically, Image Classification refers to images in which only one object appears and is analyzed. In contrast, object detection involves both classification and localization tasks, and is used to analyze more realistic cases in which multiple objects may exist in an image.

- **Conceptual Background of the Domain Problem**

  Describe the domain related concepts that you think will be useful for better understanding of the project.

  The main aim behind this project is to make a deep learning model that will classify the images just like machine learning models do for classifying classes/categories etc.

  The best example of image classification is nowadays is mask detection software that determines whether the person has wore the mask or not if not he will not be allowed to get in premises. Another example is it can be used to detect the cancer like the images of benign cells and malignant cells and it can be able to detect cancer or in same type it can be used to classify many other things.

- **Review of Literature**

  This is a comprehensive summary of the research done on the topic. The review should enumerate, describe, summarize, evaluate and clarify the research done.

The research done was only the search made for images and only on suggested website i.e., www.amazon.in

- **Motivation for the Problem Undertaken**

  Describe your objective behind to make this project, this domain and what is the motivation behind.

  The main objective behind this project of image classification using deep learning is to classify image as ladies saree or mens jeans or mens trouser.

Analytical Problem Framing

- **Mathematical/ Analytical Modeling of the Problem**

  Describe the mathematical, statistical and analytics modelling done during this project along with the proper justification.

- During the project the image was converted to array using cv2 method

- All the images was resized to common dimension of 320 length and 140 breadth

- The whole data was divided by 255 to bring on the same scale between 0 to 1

- Training data was again reshaped with (-1, width, height, channel)

- Same things were done after loading the model for prediction.

- **Data Sources and their formats**

What are the data sources, their origins, their formats and other details that you find necessary? They can be described here. Provide a proper data description. You can also add a snapshot of the data.

The images are collected using selenium you can refer to the following code snippet.

Downloading Images of Mens Trousers

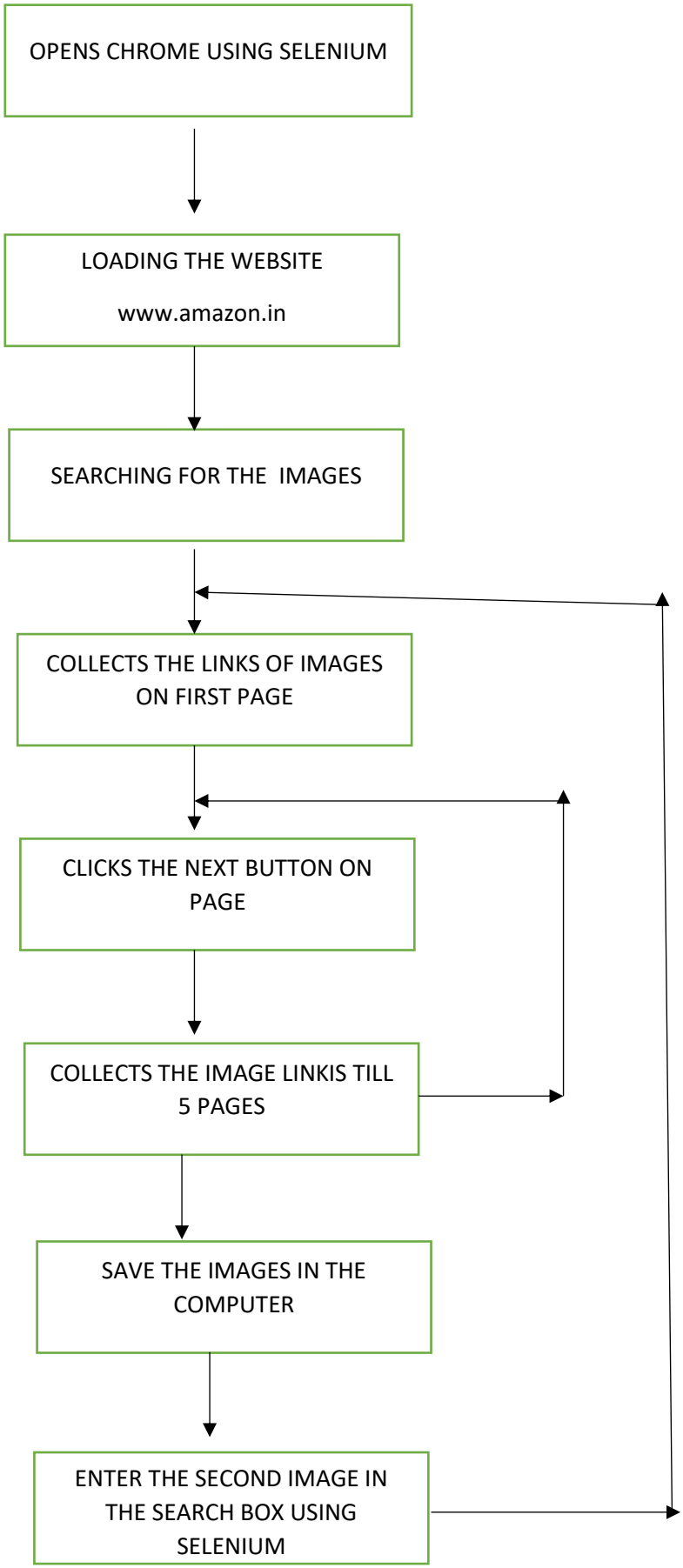
In [4]:

```
input2='mens trousers'



# entering the user's information in the search_box
search_bar=driver.find_element_by_id('twotabsearchtextbox')
search_bar.clear()
search_bar.send_keys(input2)
time.sleep(3)



# clicking the search button
search_button=driver.find_element_by_id('nav-search-submit-button')
search_button.click()
time.sleep(3)



# collecting image links from 6 pages
image_links=[]
for j in range(0,6):
    time.sleep(4)
    for i in  driver.find_elements_by_xpath('//div[@class="a-section aok-relative s-image-tall-aspect"]/img'):
        link=i.get_attribute('src')
        image_links.append(link)
    time.sleep(5)
    driver.find_element_by_xpath('//a[@class="s-pagination-item s-pagination-next s-pagination-button s-pagination-separator"]').
    time.sleep(3)

# Downloading the images in the folder
cnt=0
for i in range(len(image_links)):
    response= requests.get(image_links[i])
    file = open(r"D:\flip_robo\mens trousers\img"+str(i)+".jpg", "wb")
    file.write(response.content)
    cnt=cnt+1
print("Downloaded {0}  images of {1} " .format(cnt, input2))
```

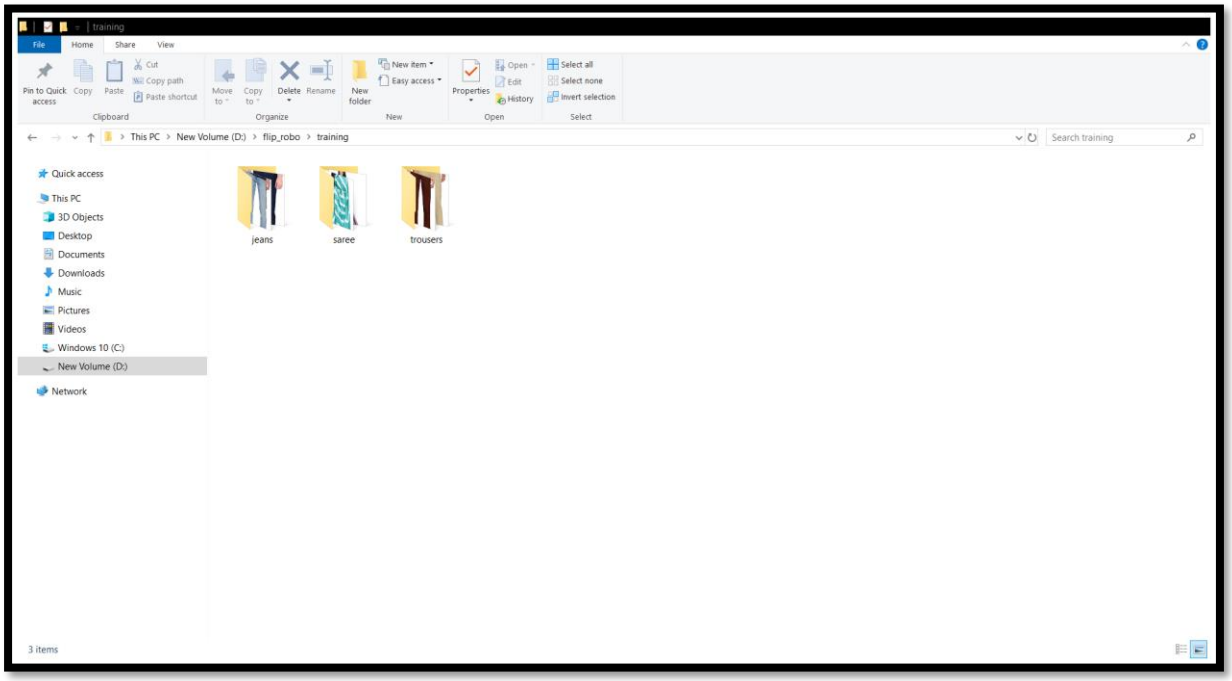
```
Downloaded 356  images of mens trousers
```

The image data is collected from the commercial website i.e., ‘amazon.in’.

The flow of collecting images was as follows


OPENS CHROME USING SELENIUM
LOADING THE WEBSITE
www.amazon.in
SEARCHING FOR THE IMAGES
COLLECTS THE LINKS OF IMAGES ON FIRST PAGE
CLICKS THE NEXT BUTTON ON PAGE
COLLECTS THE IMAGE LINKIS TILL 5 PAGES
SAVE THE IMAGES IN THE COMPUTER
ENTER THE SECOND IMAGE IN THE SEARCH BOX USING SELENIUM

The image was downloaded into single folder you can have a look at it



- **Data Preprocessing Done**

  What were the steps followed for the cleaning of the data? What were the assumptions done and what were the next actions steps over that?

  The cleaning performed in the project was not much it was just

  1) The image was converted to the array using cv2
  2) The category was converted to labels and appended in to data

creating data in required format

reading every image converting it into arrray and saving it in list along with label

In [62]:
```
# reading every image converting it into arrray and save it in list
data = []                                          # empty list to save image array and label
for category in CATEGORIES:
    folder = os.path.join(DIRECTORY, category)         # it will join the directory and category
    label=CATEGORIES.index(category)
    for img in os.listdir(folder):                       # for all the images in the folder
        img_path = os.path.join(folder, img)          # it will take all the directory of folder and  images like training/jeans/
        arr = cv2.imread(img_path, cv2.IMREAD_GRAYSCALE)  #here cv2 converts the image into array
        new_arr = cv2.resize(arr, (140,320))
        data.append([new_arr, label])
```

- **Data Inputs- Logic- Output Relationships**

  Describe the relationship behind the data input, its format, the logic in between and the output. Describe how the input affects the output.

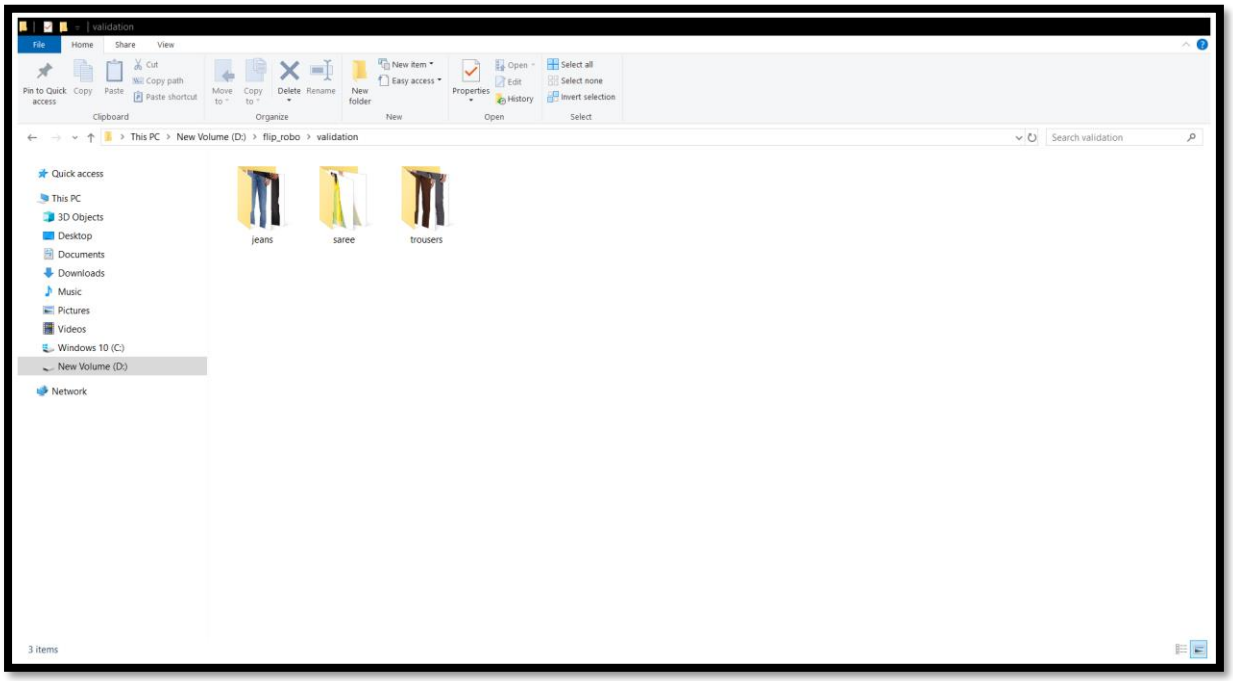
  The input was in the format of images around 900 plus images were given as input and the Output was name of items like saree, jeans, trousers.

  The input image was converted in the array and the output was converted i.e., category was converted to indexes like 1,2,3

- **State the set of assumptions (if any) related to the problem under consideration**

  Here, you can describe any presumptions taken by you.

The presumptions taken were actually not any presumptions but the process of validating data. Part of some collected data was saved in the validation folder to be passed for validating that whether the data has learned properly or not



- **Hardware and Software Requirements and Tools Used**

Listing down the hardware and software requirements along with the tools, libraries and packages used. Describe all the software tools used along with a detailed description of tasks done with those tools.

Tensorflow Environment

The Project was created in the Tensorflow environment to open Tensorflow Environment

Anaconda Prompt

↓

conda activate tf2

(my tensorflow environment is names as tf2)

↓

open Jupyter notebook

Jupyter Notebook is an open-source web application that allows to create and share document that contain live code, equations, visualiazations and narrative text.

we used jupyter notebook for importing libraries, loading dataset, data cleaning and transformation, numerical simulation,statistical modelling, visualization of data, building the models and much more

Numpy is a Python library used for working with arrays. It also has functions for working in domain of linear

algebra, fourier transform, and metrics. NumPy was created in 2005 by Travis Oliphant.

In our project we used numpy for various things like for converting negative into positive values we used absolute abs() function. Besides min,max ,Iqr are also derived from Numpy library.

Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy. Using matplotlib library we can plot the images of image data

Model/s Development and Evaluation

- **Identification of possible problem-solving approaches (methods)**

  Describe the approaches you followed, both statistical and analytical, for solving of this problem.

  To classify the images I used Convolutional Neural Network.

  The name convolutional neural network indicates the network employ a mathematical operation called convolution. convolutional network are a specialised type of neural network that use convolution in place of general matrix multiplication in at least one of their Layers

  CNN is able to successfully captured the special feature of an image to the application of relevance filters in other words the network can be trained to understand the superstition of image better

  A Convolutional Neural Network (ConvNet/CNN) is a Deep Learning algorithm which can take in an input image, assign importance (learnable weights and bias) to various aspects/objects in the image and be able to differentiate

- **Testing of Identified Approaches (Algorithms)**

Listing down all the algorithms used for the training and testing.

1) To read the directory-
    i) os library is used
2) To load the image
    i) cv2 library is used
3) To Process the image
    i) ImageDataGenerator library is used it is imported from the tensorflow.keras.preprocessing.image

4) To Create the Models-
    i) Sequential model was used. It was imported from tensorflow.keras.models
5) To Create Layers
    i)Conv2D,
    ii)MaxPooling2D,
    iii)Dense,
    iv)Flatten
    these above layers were imported from tensorflow.keras,layers

- **Run and Evaluate selected models**

  Describe all the algorithms used along with the snapshot of their code and what were the results observed over different evaluation metrics.

  The algorithm used was CNN along with sequential model in the sequential model the activation function used was ‘relu’ and at last as it was multiclass categorical classification so at last ‘softmax’ activation function was used and during the training of neurals maxpooling was used as pooling process.
  The optimizer used was adam, and loss used function used was sparse_categorical_crossentropy and for metrics accuracy was used

In [23]:
```
# importing necessary libraries
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Conv2D, MaxPooling2D, Dense, Flatten
```

In [24]:
```
# creating neural network
model = Sequential()

model.add(Conv2D(64, (3,3), activation = 'relu'))
model.add(MaxPooling2D((2,2)))

model.add(Conv2D(64, (3,3), activation = 'relu'))
model.add(MaxPooling2D((2,2)))

model.add(Flatten())

model.add(Dense(128, input_shape = x.shape[1:], activation = 'relu'))

model.add(Dense(3, activation = 'softmax'))
```

In [25]:
```
# compiling model
model.compile(optimizer='adam',
              loss='sparse_categorical_crossentropy',
              metrics=['accuracy'])
```

In [26]:
```
# creating model by 30 iterations
model.fit(x, y, epochs=30, validation_split=0.1)
```

- **Key Metrics for success in solving problem under consideration**

  What were the key metrics used along with justification for using it? You may also include statistical metrics used if any.

  The metrics used for justification was accuracy and loss, validation accuracy and validation loss.
  For the best model it is necessary that the accuracy should be close to 1 and loss should be close to the zero. It was observed that as the number of epochs increased the accuracy starts to reach near to one and in the epoch 23 it finally reached 1 and the loss was decreasing with the increasing number of epochs and at the epoch 30/30 the loss recorded was 0.0016

  The validation accuracy also recorded was 0.88 at the epoch 30/30 at epoch 1/30 it was 78.

```
28/28 [==============================] - 111s 4s/step - loss: 0.1447 - accuracy: 0.9594 - val_loss: 0.5637 - val_accuracy: 0.87
88
Epoch 15/30
28/28 [==============================] - 115s 4s/step - loss: 0.1354 - accuracy: 0.9526 - val_loss: 0.5216 - val_accuracy: 0.88
89
Epoch 16/30
28/28 [==============================] - 115s 4s/step - loss: 0.1393 - accuracy: 0.9459 - val_loss: 0.5457 - val_accuracy: 0.86
87
Epoch 17/30
28/28 [==============================] - 119s 4s/step - loss: 0.0612 - accuracy: 0.9797 - val_loss: 0.5695 - val_accuracy: 0.88
89
Epoch 18/30
28/28 [==============================] - 130s 5s/step - loss: 0.0387 - accuracy: 0.9887 - val_loss: 0.7335 - val_accuracy: 0.86
87
Epoch 19/30
28/28 [==============================] - 120s 4s/step - loss: 0.0284 - accuracy: 0.9966 - val_loss: 0.5734 - val_accuracy: 0.88
89
Epoch 20/30
28/28 [==============================] - 116s 4s/step - loss: 0.0202 - accuracy: 0.9955 - val_loss: 0.6414 - val_accuracy: 0.86
87
Epoch 21/30
28/28 [==============================] - 112s 4s/step - loss: 0.0153 - accuracy: 0.9989 - val_loss: 0.6220 - val_accuracy: 0.89
90
Epoch 22/30
28/28 [==============================] - 113s 4s/step - loss: 0.0124 - accuracy: 0.9989 - val_loss: 0.6961 - val_accuracy: 0.88
89
Epoch 23/30
28/28 [==============================] - 110s 4s/step - loss: 0.0077 - accuracy: 1.0000 - val_loss: 0.6831 - val_accuracy: 0.88
89
Epoch 24/30
28/28 [==============================] - 109s 4s/step - loss: 0.0050 - accuracy: 1.0000 - val_loss: 0.6801 - val_accuracy: 0.89
90
Epoch 25/30
28/28 [==============================] - 111s 4s/step - loss: 0.0039 - accuracy: 1.0000 - val_loss: 0.7715 - val_accuracy: 0.87
88
Epoch 26/30
28/28 [==============================] - 113s 4s/step - loss: 0.0035 - accuracy: 1.0000 - val_loss: 0.7259 - val_accuracy: 0.88
89
Epoch 27/30
28/28 [==============================] - 133s 5s/step - loss: 0.0027 - accuracy: 1.0000 - val_loss: 0.7220 - val_accuracy: 0.88
89
Epoch 28/30
28/28 [==============================] - 113s 4s/step - loss: 0.0022 - accuracy: 1.0000 - val_loss: 0.8025 - val_accuracy: 0.88
89
Epoch 29/30
28/28 [==============================] - 98s 3s/step - loss: 0.0018 - accuracy: 1.0000 - val_loss: 0.7963 - val_accuracy: 0.888
9
Epoch 30/30
28/28 [==============================] - 116s 4s/step - loss: 0.0016 - accuracy: 1.0000 - val_loss: 0.8081 - val_accuracy: 0.88
89
```

- **Visualizations**

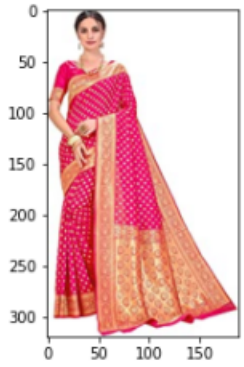
Mention all the plots made along with their pictures and what were the inferences and observations obtained from those. Describe them in detail.

If different platforms were used, mention that as well.

As it was image classification project so there were no plots or graphs made but yes the image were loaded in the jupyter notebook you can look at following snippets

In [4]:
```
# loading single image to check whether data is properly imported or not
img=image.load_img('D:/flip_robo/training/saree/img2.jpg')
plt.imshow(img)
cv2.imread('D:/flip_robo/training/saree/img2.jpg').shape
```

Out[4]: (320, 188, 3)



saree images has a dimension of 320 length , 188 width and it is a RGB image

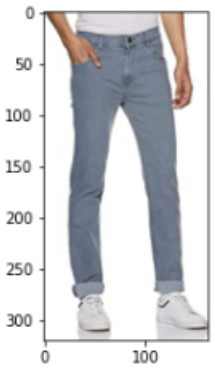
In [5]:
```
# loading single image to check whether data is properly imported or not
img=image.load_img('D:/flip_robo/training/trousers/img0.jpg')
plt.imshow(img)
cv2.imread('D:/flip_robo/training/trousers/img0.jpg').shape
```

Out[5]: (320, 141, 3)

trouser images has a dimension of 320 length , 141 width and it is a RGB image

In [6]:
```
# loading single image to check whether data is properly imported or not
img=image.load_img('D:/flip_robo/training/jeans/img0.jpg')
plt.imshow(img)
cv2.imread('D:/flip_robo/training/jeans/img0.jpg').shape
```

Out[6]: (320, 162, 3)



jeans images has a dimension of 320 length , 141 width and it is a RGB image

- # Interpretation of the Results

  Give a summary of what results were interpreted from the visualizations, preprocessing and modelling.

  Earlier while building the model when 1) the epochs were set to 5 at that time the accuracy was at 78% and the loss was around 0.5 something but as soon as the epochs were set to 30 at the time the accuracy increased to 1 and loss to 0.0016

  2)The image were also set to grayscales the result was not so desirable so it was decided to keep it in RGB format

CONCLUSION

- ## Key Findings and Conclusions of the Study

  Describe the key findings, inferences, observations from the whole problem.

- ## Learning Outcomes of the Study in respect of Data Science

  List down your learnings obtained about the power of visualization, data cleaning and various algorithms used. You can describe which algorithm works best in which situation and what challenges you faced while working on this project and how did you overcome that.

  The overall thing I learnt is the

  More the image is cleaned perfect is the result,

  More the number of layers more is the number of neurons,

  More the number of epochs more the model learns.

- # Limitations of this work and Scope for Future Work

What are the limitations of this solution provided, the future scope? What all steps/techniques can be followed to further extend this study and improve the results.

Limitations

There was no limitation found in the project as it was simple short and easy project

Future Scope

In Future Scope we can add more categories of images to classify more items