OPTIMIZING PRODUCT INFORMATION : IMAGE PROCESSING IN RETAIL INDUSTRY

Author Names:- Rahul Shriram Athavale And Yashita Digambar Parab

College Name :- ASM's Institute of Management & Computer Studies,Thane

Abstract:-

With the growth of the retail industry day by day, there is also an increase in the number of users. As we can see, a number of products are sold via the internet.

Considering Amazon's Ecommerce platform,an average of 1.3 million products is added each Day. Added such a large number of data manually may lead to inconsistency of data related to particular products. Consequently, image processing plays a significant role in the detection of attributes from images, such as the type of neck, the type of sleeves, the type of color, and the type of pattern.

Keywords - Face Detection, Algorithm, CNN, Feature Detection

INTRODUCTION:-

The primary objective of this article is to present a solution to address the issue of inconsistent data pertaining to garments on ecommerce platforms. Taking the instance of a casual shirt, which encompasses information regarding its color, pattern, sleeves, and neck type, there are instances where the displayed information on the website may be inaccurate. For instance, the sleeves type might be labeled as 'full sleeves' when it should actually be 'half sleeves'. To avoid such types of incorrect data, we can use image driven attribute detection system with the help of image processing and machine learning will play an important role to correctly identify it.

II. PROCEDURES FOR ATTRIBUTE DETECTION

A. Removal of Gradient Background ColorSTEP 1: Edge Detection

When aiming to eliminate the gradient background from an image, the initial step involves detecting the outline of the specific object. This detection enables us to determine the precise pixel boundary where the color replacement should occur. To accomplish this, the widely recognized Canny Edge Detection algorithm is employed. The Canny edge algorithm is renowned for its effectiveness in detecting all edges within an image. Its superiority stems from its systematic approach, as it follows a defined set of procedures to attain the desired output. Consequently, the Canny edge detection algorithm is considered the most suitable choice for accurate edge detection.

● Noise reduction can be achieved by employing the Gaussian filter to apply a blur to the image. This blurring operation serves to eliminate edges that are not well-defined and remove extraneous points outside the object region. By effectively blurring the image, the Gaussian filter helps to enhance the clarity and focus of the desired object while reducing unwanted noise.

● The process of gradient calculation plays a pivotal role in determining the intensity and direction of edges present in an image. By accurately calculating the gradients, we gain valuable information about the sharpness and orientation of the edges. Once the fine edges have been successfully detected, the application of the Sobel filter comes into play. This filter is specifically employed to blur the edges, resulting in a more refined and smoothed representation of the edge boundaries.

● Non-Maximum Suppression is a straightforward procedure that involves traversing each point in the gradient magnitude grid and identifying pixels with the highest value in the edge directions. This guideline ensures that only the most prominent and significant pixels along the edge orientations are retained, while suppressing the rest. By applying this technique, we can effectively highlight and preserve the essential edge features while reducing redundancy in the image representation.

● Hysteresis Thresholding is a two-step process that aims to categorize pixels into three types: strong, weak, and non-relevant. Strong pixels are those with an intensity value high enough to confidently contribute to the final edge detection. Weak pixels, on the other hand, possess an intensity value that falls below the threshold for strong pixels but is still significant enough to not be considered irrelevant for edge detection. Any remaining pixels are classified as non-relevant for the edge detection process. This approach ensures that pixels are appropriately classified based on their intensity levels, allowing for a more accurate and refined identification of edges.

STEP 2: Contour Detection

Contour detection is a method that can be succinctly described as a curve connecting all the points with a consistent color or intensity, including the image border. Contours serve as valuable tools for shape analysis and object detection and recognition. OpenCV, a computer vision library, provides a feature called "Find Contours" that aims to locate contours within an image. However, to achieve accurate contour detection, certain preprocessing techniques must be applied to the image. In the example below, the morphology feature is employed using the CV_MOP_OPEN and CV_MOP_CLOSE techniques to refine the contours. Subsequently, the "Find Contours" function is utilized to identify and print the detected contours.

➢ Dilation is a method that involves convolving a specific kernel, such as a square or a circle, with the image. This kernel contains an anchor point that corresponds to its center. By overlaying this kernel onto the image, the highest pixel value within the kernel's range is computed. The resulting value is then assigned to the anchor point, effectively replacing the original pixel value at that location. This operation aims to enhance the size of bright regions and consequently reduce the overall size of the image. By selectively expanding the areas of interest, dilation can be used to accentuate specific features or modify the image structure.

➢ Erosion is a method that closely resembles dilation, but with a different approach. In erosion, the calculated pixel value is the minimum value within the kernel's range, as opposed to the maximum value in dilation. The image is then updated by replacing the original pixel value at the anchor point with this minimum value. As a result, erosion contributes to the growth in size of dark regions while diminishing the size of bright regions. By selectively eroding specific areas, this operation can be used to modify the image structure and highlight dark features while suppressing bright regions.

➢ Masking comes into play after the processes of erosion and dilation have been applied, resulting in the retention of objects with well-defined edges. The next step is to transfer this object onto a transparent or white background, facilitating the initial stage of gradient removal. To achieve this, a new canvas with specific dimensions is created, featuring a white background. The object image is then redrawn onto this new canvas, effectively placing the object in the desired context. This process ensures that the object is isolated and ready for further manipulation, such as the removal of gradients, as it is now situated on a uniform background.

Example:

Gradient Image   Without Gradient

B. Crop Specific Part from Images(Neck, Sleeves)

To detect and crop specific parts from an image, such as the neck and sleeves, a systematic process is followed. Initially, the face of the model within the image needs to be detected in order to perform accurate distance calculations for all the parts. OpenCV provides a face detection algorithm that can be utilized for this purpose. Once the face is detected, a rectangle is drawn around it, enabling the extraction of the upper and lower coordinates of the face. With the lower coordinates of the face obtained, a mathematical formula is applied to calculate the distance of the neck from these coordinates. The same procedure is followed to determine the distances of the right and left hand from the neck. By employing these calculations, the specific parts of interest can be accurately cropped from the image, facilitating further analysis and attribute detection.

Figures:

Original Image   Neck Crop   Pattern Crop

C. Conversion of crop images to black and white

In order to optimize the processing power of the GPU and ensure accurate attribute detection, it is necessary to make certain adjustments to the cropped images. One crucial step is to change the background to black, while keeping the outline of the object in white. This transformation will result in the images being displayed as follows:

The object within each image will be prominently highlighted, as its outline will be represented by white pixels against a black background. This visual representation allows for enhanced contrast and clarity, facilitating more accurate attribute detection algorithms. By standardizing the background and emphasizing the object's outline, the images become more suitable for further analysis and processing.
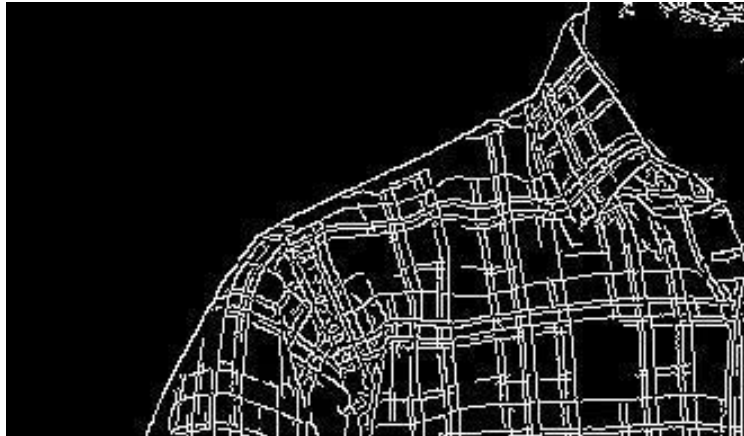
D. Classification and Feature Detection

The final step involves the detection of features in order to extract attribute details from the images. Numerous open-source tools are available for this purpose, including PyTorch, TensorFlow, and others. In this case, we will utilize Keras for attribute derivation. Keras is a widely-used framework that provides a high-level interface for building and training deep learning models. By leveraging the capabilities of Keras, we can efficiently extract relevant features from the images and derive attribute information. Its user-friendly API and extensive documentation make it a suitable choice for attribute detection tasks.

Step 1: In classification tasks, having a training dataset and a test dataset is crucial. The training dataset should be larger than the test dataset, as it allows the model to learn a wider range of patterns and reduces the risk of overfitting. The test dataset should be representative of real-world data to evaluate the model's performance accurately.

Step 2: The convolution layer's main purpose is to extract features from images. It does this by analyzing small square regions of input data called filters. The layer maintains the spatial relationship between pixels, allowing it to capture patterns and details effectively. The convolution operation produces feature maps that highlight specific visual characteristics in the image. Overall, convolutional layers play a crucial role in learning hierarchical representations of images in deep learning models.

Step 3: Pooling is a technique used to reduce the dimensionality of feature maps while retaining important information. One common pooling method is Max Pooling, where a spatial neighborhood (e.g., a 2x2 window) is defined, and the maximum value within that window is selected from the feature map. Alternatively, the average or sum of the values within the window could be taken. However, Max Pooling has been observed to yield better results in practice. By performing pooling operations, the spatial size of the feature maps is reduced, leading to a more compact representation while preserving the most significant features.

Step 4: Once all the necessary steps have been completed, the next phase is training the model. The training process duration depends on the hardware used. Training with a powerful video card and ample RAM can be accomplished in less than an hour. However, if training is conducted on a regular

CPU, it may take significantly longer. The model's precision tends to improve with an increasing number of epochs. By allowing the model to iterate over the training data multiple times, it can gradually learn and refine its predictions, resulting in improved accuracy. The number of epochs chosen for training depends on the complexity of the problem and the available computational resources.

E. Conversion of crop images to black and white

Directly passing cropped images to the convolution layer can boost GPU processing power but may result in incorrect attribute detection. Pre-processing steps like   re-sizing, normalization, and background adjustment should be applied to ensure consistent image quality and improve detection accuracy. This allows the convolution layer to effectively analyze standardized features and extract accurate attribute information.

III. CONCLUSION

In conclusion, this technology offers significant benefits to operational personnel responsible for uploading product data on websites. By effectively classifying images and generating an Excel file, it ensures accurate data insertion into the database. Consequently, this application provides a reliable solution to address the issue of data inconsistency related to clothing items.

IV. REFERENCES

1) https://docs.opencv.org/master/d9/df8/tutorial_root.html

2) https://keras.io/

3) https://www.geeksforgeeks.org/image-processing