2024

6(1), 57-74

ShoppingTotal: A Mobile Application Utilizing Assisted Rekognition Algorithm for Intelligent Price Detection from Shelf Label Images

Zuhal Can*[1]

Keywords

Mobile application
Shopping monitoring
Amazon Rekognition
Text detection
Image filters
Price tags

Received
Apr 19, 2024

Accepted
June 29, 2024

Published
June 30, 2024

Article Type
Research Paper

Abstract

ShoppingTotal is a mobile application for monitoring the shopping budget through shelf label images. Using the ShoppingTotal application, shoppers capture the shelf label image of the product to obtain the product information and view the total amount of the current shopping and the history of the previous shopping lists. For the ShoppingTotal application, the Assisted Rekognition algorithm is developed based on Amazon Rekognition's text detection service for extracting product information from label images. The FourGroceries dataset is collected for evaluating the performance of the Assisted Rekognition algorithm over original, single-filtered, and multi-filtered images based on the image filters under the categories of sharpness, blurriness, brightness, temperature, and color. According to experiments on the FourGroceries dataset and the Amazon Rekognition service, the average price detection confidence results are 76.49% with the Assisted Rekognition algorithm and 20.94% without the Assisted Rekognition algorithm. The Assisted Rekognition algorithm's performance is found to be better on filtered images than on original images, with 89.25% price detection confidence. By applying appropriate single or multiple image filters on the FourGroceries dataset, the Assisted Rekognition algorithm achieves extracting the correct price values from all experimental dataset images.

ShoppingTotal: Raf Etiketi Görüntülerinden Akıllı Fiyat Tespiti için Desteklenmiş Rekognition Algoritması Kullanan Mobil Uygulama

Anahtar Sözcükler

Mobil Uygulama
Alışveriş İzleme
Amazon Rekognition
Metin algılama
Görüntü filtreleri
Fiyat etiketleri

Gönderim Tarihi
20 Nisan 2024

Kabul Tarihi
29 Haziran 2024

Yayın Tarihi
30 Haziran 2024

Makale Türü
Araştırma Makalesi

Öz

ShoppingTotal, raf etiketi görselleri aracılığıyla alışveriş bütçesini izlemeye yönelik bir mobil uygulamadır. Alışveriş yapanlar, ShoppingTotal uygulamasını kullanarak ürünün raf etiketi görselini yakalayarak ürün bilgilerine ulaşabilir, mevcut alışverişin toplam tutarını ve önceki alışveriş listelerinin geçmişini görüntüleyebilir. ShoppingTotal uygulaması için Destekli Rekognition algoritması, Amazon Rekognition'ın etiket görüntülerinden ürün bilgilerini çıkarmaya yönelik metin algılama hizmetini temel alarak geliştirilmiştir. FourGroceries veri kümesi, Destekli Rekognition algoritmasının performansını, keskinlik, bulanıklık, parlaklık, sıcaklık ve renk kategorileri altındaki görüntü filtrelerine dayalı olarak orijinal, tek filtreli ve çoklu filtreli görüntüler üzerinde değerlendirmek için toplanır. FourGroceries veri seti ve Amazon Rekognition hizmeti üzerinde yapılan deneylere göre ortalama fiyat tespit güven sonuçları, Assisted Rekognition algoritması ile %76,49, Assisted Rekognition algoritması olmadan ise %20,94'tür. Destekli Rekognition algoritmasının performansının, %89,25 fiyat tespit güveniyle, filtrelenmiş görüntülerde orijinal görüntülere göre daha iyi olduğu bulundu. Destekli Rekognition algoritması, FourGroceries veri kümesine uygun tekli veya çoklu görüntü filtreleri uygulayarak, tüm deneysel veri kümesi görüntülerinden doğru fiyat değerlerinin çıkarılmasını sağlar.

Atıf: Can, Z. (2024). ShoppingTotal: Raf etiketi görüntülerinden akıllı fiyat tespiti için desteklenmiş rekognition algoritması kullanan mobil uygulama. *Bilgi ve İletişim Teknolojileri Dergisi, 6*(1), 57-74. https://doi.org/10.53694/bited.1470771

Cite: Can, Z. (2024). Shoppingtotal: A mobile application utilizing assisted rekognition algorithm for intelligent price detection from shelf label images. *Journal of Information and Communication Technologies, 6*(1), 57-74. https://doi.org/10.53694/bited.1470771

***Sorumlu Yazar/Corresponding Author**

[1] Eskişehir Osmangazi University, Faculty of Engineering and Architecture, Department of Computer Engineering, Eskişehir, Turkey, zcan@ogu.edu.tr, https://orcid.org/0000-0002-6801-1334

Introduction

Due to unintended purchases, exceeding the shopping budget is a common problem for shoppers. Even if shoppers intend to buy the stuff they came for in the first place, they may purchase additional items during shopping. Splurging on extra purchases causes shoppers to exceed their shopping budget at checkout. A solution for this unplanned purchase problem is monitoring the shopping budget using an assistant shopping application, which gives shoppers control over their shopping budget.

ShoppingTotal is a mobile application developed to monitor the shopping cart total based on shelf labels. ShoppingTotal processes the shelf label images, extracts the price from shelf labels with the shopper confirmation and calculates the shopping total as items added to their shopping cart. By reading information from shelf labels, the ShoppingTotal application prevents shoppers from coming up with an unexpected price at checkout.

ShoppingTotal interacts with the Amazon Rekognition API that detects the image text with a confidence value. The Assisted Rekognition algorithm is developed for the ShoppingTotal application that extracts price values fully or partially from detected texts received from the Amazon Rekognition.

To assess the Assisted Rekognition's performance, shelf label images were collected from various groceries during the year 2022 under the FourGroceries dataset. FourGroceries is an image dataset of shelf labels collected from four different groceries for research purposes that consists of 21 label images for each grocery, a total of 84 images. Based on the confidence value, this study evaluates the Assisted Rekognition algorithm's performance on the FourGroceries dataset's original, single-filtered, and multi-filtered images. The applied filters on images can affect the performance of the Assisted Rekognition algorithm. For assessing the Assisted Rekognition's performance on filtered images, several filters are selected within the categories of sharpness, blurriness, brightness, temperature, and color.

The ShoppingTotal application is developed based on the text detection feature of the Amazon Rekognition service (*Amazon Rekognition*, n.d.) for information detection from shelf labels. Amazon Rekognition has several other features for object detection and image classification. Studies on Amazon Rekognition include image tagging (Kuang et al., 2021), face identification (Ali et al., 2022)(Liu & Wilkinson, 2020), emotion detection (Yang et al., 2021), and image captioning (Leotta et al., 2022).

There are similar technologies for text detection, including Amazon Textract, Google Cloud Vision API, Microsoft Azure Cognitive Services, and Tesseract OCR (Optical Character Recognition). These technologies are developed to analyze and detect text on various document types, such as scanned documents or images. Server-based engines such as Textract are found to result more accurately than standalone libraries like Tesseract, especially on noisy documents (Hegghammer, 2022). Amazon Rekognition's performance resulted better in recognizing blurry and stylized text on colored backgrounds than Amazon Textract.

Various mobile applications are developed for organizing grocery shopping. Out of Milk is a grocery shopping planner application that scans barcodes and gathers information based on the available information in the local database of the service used for the barcodes (*Out of Milk - The Grocery Shopping List App*, n.d.). Another barcode scanner application ShopSavvy is a shopping assistant application that retrieves price information based on the barcode information (*ShopSavvy*, n.d.). Scan&Shop is a shopping assistant application that allows users to scan barcodes and compare prices (*Scan&Shop*, n.d.). The iCheck application scans barcode images to calculate the total price of the shopping list (AlWadani & AlOtaibi, 2019). None of these applications are developed dedicated explicitly to shelf label information detection using the Amazon Rekognition's text detection feature.

Contributions of this paper are summarized below:

- The ShoppingTotal mobile application is developed to monitor the shopping budget by processing the shelf label images.

- The Assisted Rekognition algorithm is developed to improve the price detection performance of the ShoppingTotal application.

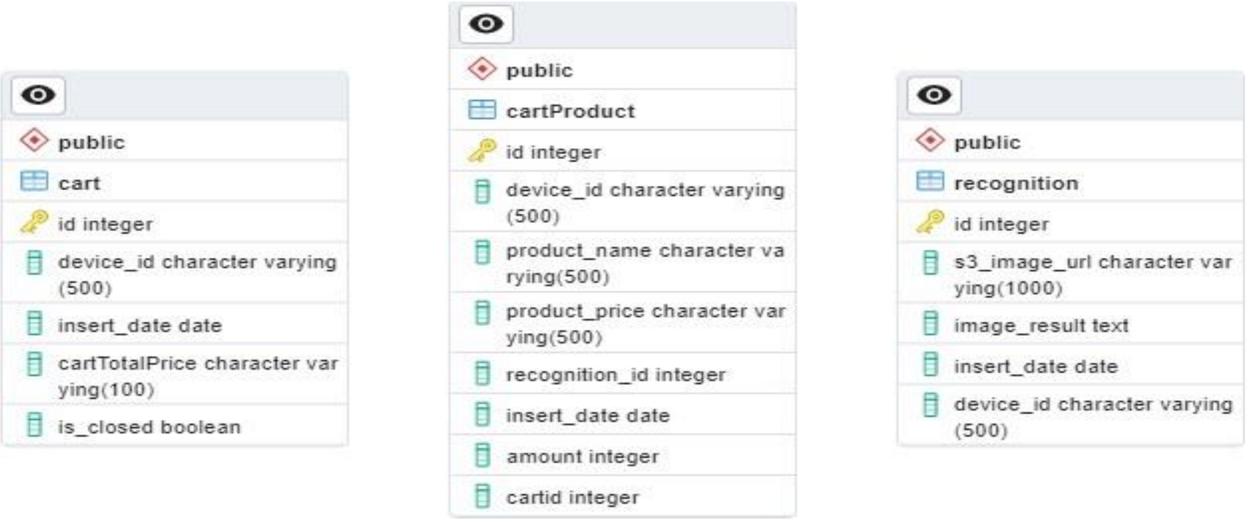
- FourGroceries dataset is collected for evaluating the performance of the Assisted Rekognition algorithm on original and filtered images.

Method

1. ShoppingTotal Application

The ShoppingTotal mobile application is developed on Android Studio in the Kotlin language. ShoppingTotal's web Application Programming Interface (API) is developed on Visual Studio by C# language to send requests to the Amazon Rekognition server to detect and filter the price text on shelf label images of grocery products. The web API runs on the EC2 instance (*Amazon Elastic Compute Cloud*, n.d.) in the Amazon Web Services (AWS) Cloud. As a REST Client, Retrofit provides the connection between the web API and Kotlin code. Captured label images are stored in the S3 Bucket (*Amazon Simple Storage Service*, n.d.) in the Amazon server and sent to Amazon Rekognition service for text detection through the web API. After processing the required information from received texts, the web API stores information on the PostgreSQL database and responds to the mobile application client with product information.

The PostgreSQL database of the ShoppingTotal application runs on the Amazon Relational Database Service (Amazon RDS) (*Amazon Relational Database Service*, n.d.). The database consists of three tables, as shown in Figure 1. The Cart table is the table that keeps the total amount of the shopping and information on whether the shopping cart is active or not. The CartProduct table is the table for the products. The Recognition table is the table that holds the image information. A shopping cart can contain more than one product. There is a 1-N relationship between the Cart and CartProduct tables and 1-1 relationship between Recognition and CartProduct tables.



public
cart
id integer
device_id character varying(500)
insert_date date
cartTotalPrice character varying(100)
is_closed boolean

public
cartProduct
id integer
device_id character varying(500)
product_name character varying(500)
product_price character varying(500)
recognition_id integer
insert_date date
amount integer
cartid integer

public
recognition
id integer
s3_image_url character varying(1000)
image_result text
insert_date date
device_id character varying(500)

Figure 1. Database tables of ShoppingTotal application

The mobile application client requests the web API to add the captured product to the shopping cart list. As a result of this request, the product is added to the database through the web API and product information is listed on the screen to be added to the cart list. Whenever a mobile application user updates the quantity of the cart products or wants to delete an item, the application client sends requests to the web API to update the database. After ending a shopping, users can view the completed shopping list within the shopping list history. The data flow on the ShoppingTotal application is shown in Figure 2.

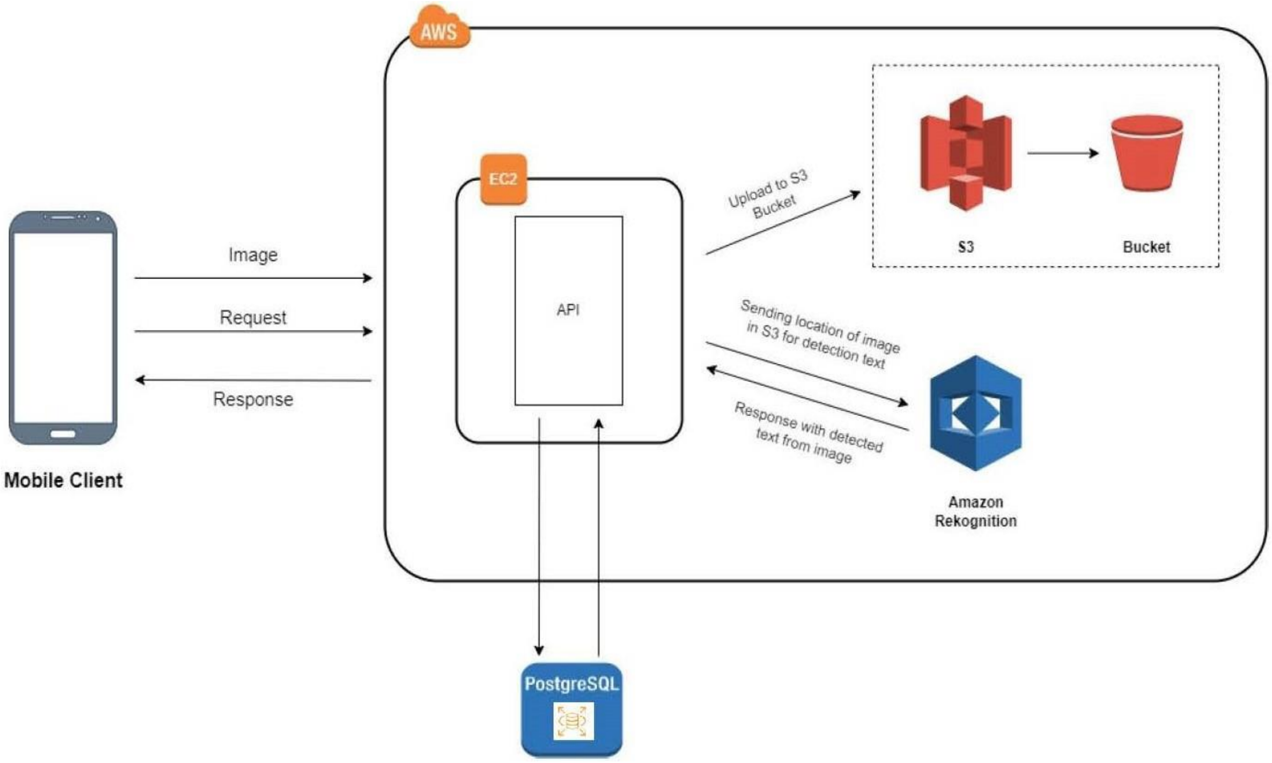


Figure 2. Data flow diagram of the ShoppingTotal application

2. The Assisted Rekognition Algorithm

The web API receives all detected texts from the Amazon Rekognition API with a confidence value. The correct price values can be received as a detected text, substring within a detected text, or partially within different detected texts.

The Assisted Rekognition Algorithm is developed for the ShoppingTotal Application to list the possible price values detected on a label image. This algorithm extracts possible price values from detected texts fully or partially. Then the algorithm keeps a record of the confidence value based on the price detection method.

Full price detection: The Assisted Rekognition Algorithm extracts the possible price value as a whole from a detected text, regardless of the punctuation mark between the integer and fractional part, and then records the confidence value as the confidence value of the detected price.

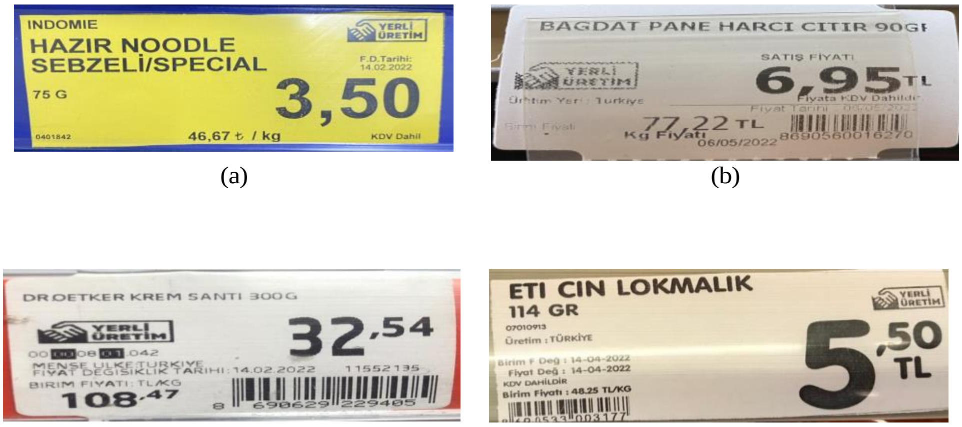
Partial price detection: If a price value is not found by the full price detection method, the algorithm extracts the integer and fractional part of the possible price value from separate detected texts and then records the confidence value as the average of the confidence values of each detected price.

3. FourGroceries Dataset

Tablo 1. FourGroceries dataset

Grocery name	Number of images	Resolution	Average image size	File format
Grocery1	21	72 dpi	987.4 KB	JPG
Grocery2	21	72 dpi	871.5 KB	JPG
Grocery3	21	96 dpi	80.6 KB	jpeg
Grocery4	21	96 dpi	71.2 KB	jpeg

For this study, the FourGroceries dataset is collected for research purposes on price detection analysis. This dataset was collected from four groceries in Turkey in 2022 by mobile phones with IOS or Android operating systems. The dataset consists of 84 images of shelf labels, 21 images from each grocery. Image properties of the dataset are given in Table 1. Example shelf label images from each grocery dataset are shown in Figure 3.



(a) (b)

(c) (d)

Figure 3. Example price tags from each grocery. (a) Grocery1. (b) Grocery2. (c) Grocery3. (d) Grocery4

4. Applied Filters on Dataset Images

There are various image filters for editing images with diverse effects. The ShoppingTotal application users can apply some image filters to the camera before capturing shelf labels to improve the price detection performance of the application. While some image filters, such as blurriness, may decrease, some filters, such as sharpness, may improve the price detection performance of the ShoppingTotal application.

This paper evaluates image filter effects on the Assisted Rekognition algorithm's price detection performance through the FourGroceries dataset. For this purpose, due to space constraints, commonly used image filters, including sharpness, blurriness, brightness, temperature, and color, are selected to evaluate the price detection performance of the algorithm. These filters and their selected parameters are given in Table 2 and explained in this section.

Table 2. Image filters

Filter type	Filters
Sharpness	Kernel: {Kernel1, Kernel2}
Blurriness	Radius: {1, 2, 3, 4, 5}
Brightness	Factor: {0.5, 1.0, 1.5, 2.0}
Temperature	Kelvin: {1000, 2000, 3000, 4000, 5000, 6000, 7000, 8000, 9000, 10000}
Color	Colors: {BGR2HLS, RGB2HLS, HSV2BGR, HSV2RGB, Lab2BGR, Lab2RGB}

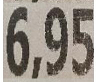
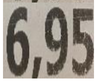
4.1. Sharpness Filters

A common method for editing image sharpness is using the Laplacian kernel for sharpening images based on intensity changes. This study uses two Laplacian kernels, as shown in Table 3. Kernel1 filter has a central positive value of nine and negative values surrounding it. The positive central value enhances the edges of the areas of high frequency, while the negative values surrounding the central value suppress neighboring pixels. Therefore, Kernel1 is selected to make a more sharpening effect on images than Kernel2, as shown in Table 4.

Table 3. Kernel values for sharpness filters

Kernel Name	Value
Kernel1	[[-1, -1, -1], [-1, 9, -1], [-1, -1, -1]]
Kernel2	[[0, -1, 0], [-1, 5, -1], [0, -1, 0]]

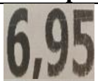
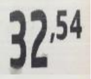
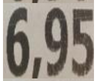
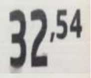
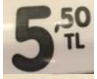
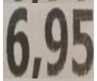
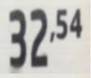
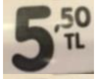
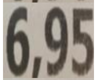
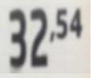
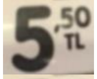
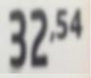
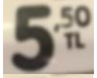
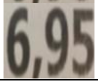
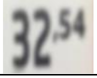
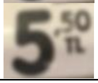
Table 4. Sharpness filter effects on images

Sharpness Filter	Grocery1 Sample	Grocery2 Sample	Grocery3 Sample	Grocery4 Sample
Kernel1				
Kernel2				
Nonfiltered				

4.2. Blurriness Filters

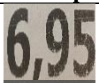
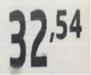
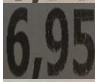
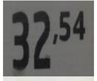
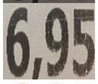
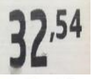
The blurriness filter smooths images by reducing high-frequency details in an image. In this study, Gaussian blurring is used as the blurring method. In Gaussian blurring, the radius is the extension value of blurring around each pixel in the image. Smaller values of radius result in a less blurring effect on an image. In this study, various values of radius up to the value of five are selected for blurring images, as shown in Table 5. A radius value of five has a more blurring effect on images than a radius value of one.

Table 5. Blurriness filter effects on images

Blurriness Filter	Grocery1 Sample	Grocery2 Sample	Grocery3 Sample	Grocery4 Sample
Nonfiltered				
Radius=1				
Radius=2				
Radius=3				
Radius=4				
Radius=5				

4.3. Brightness Filters

Table 6. Brightness filter effects on images

Brightness Filter	Grocery1 Sample	Grocery2 Sample	Grocery3 Sample	Grocery4 Sample
Nonfiltered				
Brightness (Factor=0.5)				
Brightness (Factor=1.0)				
Brightness (Factor=1.5)				
Brightness (Radius=2.0)				

The factor value of the brightness filter adjusts the overall brightness of an image by multiplying it with each pixel. As the factor value increases, an image gets brighter. Multiplication of each pixel value with a factor value greater than one results in a brighter appearance on an image, and a factor value less than one results in a darker image, as shown in Table 6.

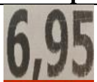
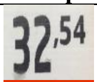
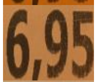
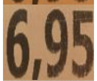
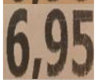
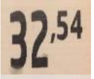
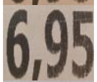
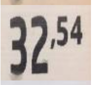
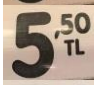
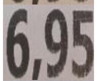
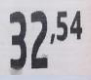
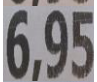
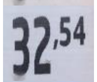
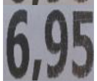
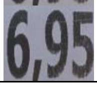
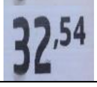
4.4. Temperature Filters

Adjusting R (Red), G (Green), and B (Blue) values ranging from 0 to 255 allows applying a temperature filter into an image corresponding to a temperature value in Kelvin. An image can appear in warmer or cooler colors based on the adjustments to RGB values. The correlation between RGB values and temperature can vary depending on color calibration. This study applies the temperature filters on images according to the corresponding RGB values, as shown in Table 7, based on the RGB to color temperature study (Siess, n.d.). Temperature effects on sample images are shown in Table 8.

Table 7. Temperature filters conversion values for R, G, B

Temperature (in Kelvin)	Conversion values for R, G, B
1000	255, 56, 0
2000	255, 137, 18
3000	255, 180, 107
4000	255, 209, 163
5000	255, 228, 206
6000	255, 243, 239
7000	245, 243, 255
8000	227, 233, 255
9000	214, 225, 255
10000	207, 218, 255

Table 8. Temperature filter effects on images

Temperature Filter	Grocery1 Sample	Grocery2 Sample	Grocery3 Sample	Grocery4 Sample
Nonfiltered				
Temperature (1000 K)				
Temperature (2000 K)				
Temperature (3000 K)				
Temperature (4000 K)				
Temperature (5000 K)				
Temperature (6000 K)				
Temperature (7000 K)				
Temperature (8000 K)				
Temperature (9000 K)				
Temperature (10000 K)				

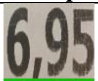
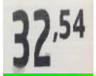
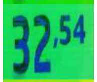
4.5 Color Filters

Various color models are developed based on the adjustments of image factors. For example, BGR represents colors by combining blue, green, and red components. Similarly, by the RGB model, each image pixel consists of three values representing the intensity of blue, green, and red channels. In this study, several color models are selected as color filters. Descriptions of the abbreviations of the selected filters are listed in Table 9. Among these filters, the HLS filter represents colors based on their hue, lightness, and saturation, whereas HSV represents colors based on their hue, saturation, and values. Hue represents the color tone, lightness refers to the perceived brightness, saturation represents the intensity of the color, and the value represents the brightness. The lab is a color model designed to imitate human perception. The effects of these selected color filters on sample images are shown in Table 10.

Table 9. Color filter abbreviations and descriptions

Abbreviation	Description
BGR	Blue Green Red
RGB	Red Green Blue
HLS	Hue Lightness Saturation
HSV	Hue Saturation Value
Lab	Lightness component L, and color components a and b

Table 10. Color filter effects on images

Color Filter	Grocery1 Sample	Grocery2 Sample	Grocery3 Sample	Grocery4 Sample
Nonfiltered				
BGR2HLS				
RGB2HLS				
HSV2BGR				
HSV2RGB				
Lab2BGR				
Lab2RGB				

Findings

1. Application Results

On the ShoppingTotal application, users can navigate to the My List, Camera, and My Cart pages from any page. The ShoppingTotal Application starts the image-capturing process after the user pushes the button with the camera icon. The screenshot of the Camera page is shown in Figure 4. After the user captures a shelf label, the label image is processed by the web API, and the application lists possible prices of the shelf label image for the user to select the correct price value, as shown in Figure 5.

The product information is added to the My Cart list when the user pushes the ADD TO CART button, as shown in Figure 6. The total price amount of the shopping cart list is given on the My Cart page. The user can delete the listed item or update the number of items on this page. The End Shopping button on the page ends the shopping list updating process for the current shopping cart. The user can list old shopping lists on the Old Shoppings page by pushing the My List button, as shown in Figure 7. On this page, users can view or delete old shopping lists.

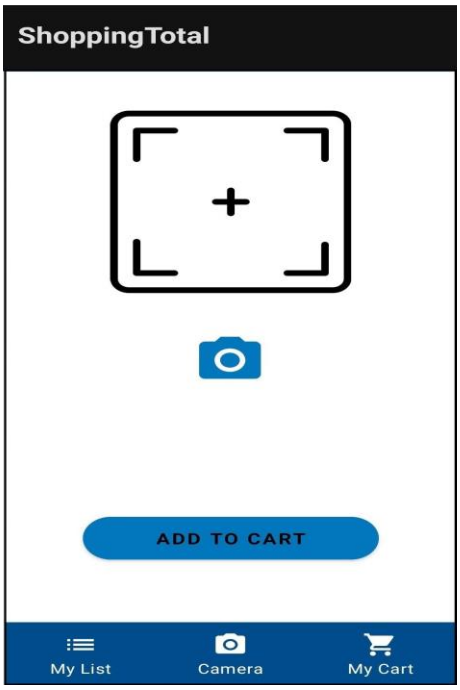
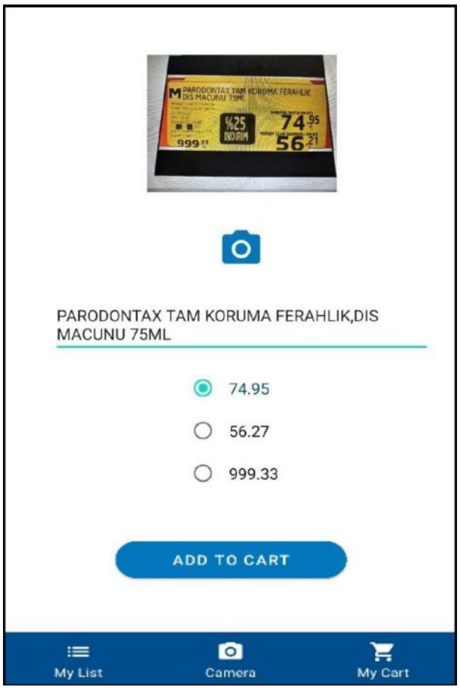


Figure 4. Camera page of the ShoppingTotal application



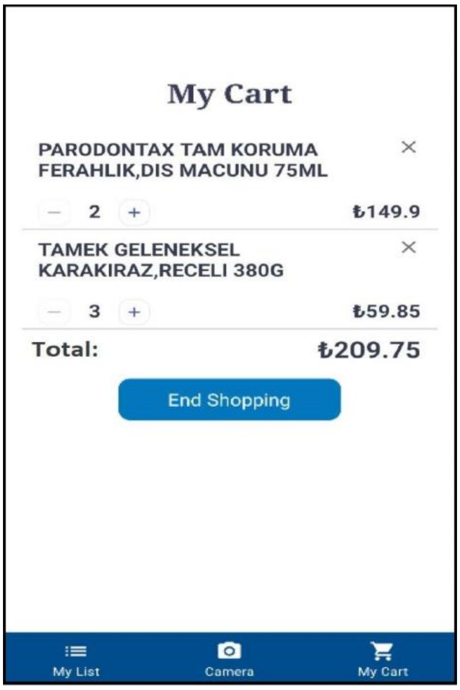
PARODONTAX TAM KORUMA FERAHLIK,DIS MACUNU 75ML

● 74.95

○ 56.27

○ 999.33

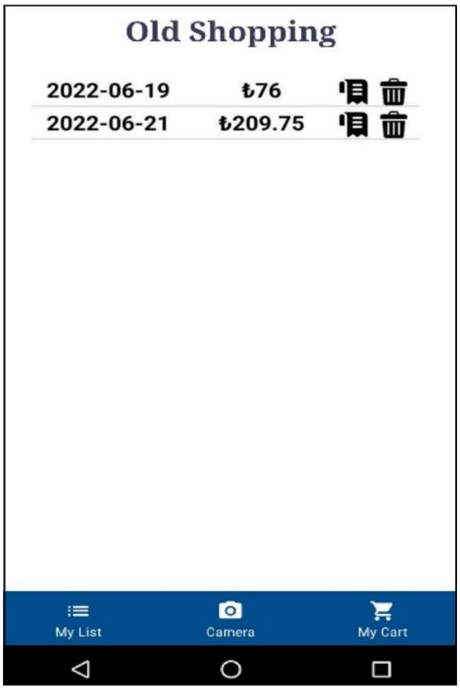
Figure 5. The ShoppingTotal application lists the possible values of an item extracted from the item’s shelf label



My Cart

PARODONTAX TAM KORUMA FERAHLIK,DIS MACUNU 75ML	
− 2 +	₺149.9
TAMEK GELENEKSEL KARAKIRAZ,RECELI 380G	
− 3 +	₺59.85
Total:	₺209.75

Figure 6. My Cart page of the ShoppingTotal application



Old Shopping

2022-06-19	₺76	
2022-06-21	₺209.75	

Figure 7. My List page of the ShoppingTotal application

2. Evaluation of the Assisted Rekognition Algorithm

2.1. Single-Filter Effects

The price detection performance of the Assisted Rekognition algorithm is evaluated based on the accurate price values of the 84 images of the FourGroceries dataset. Figure 10 demonstrates the benefit of the Assisted Rekognition algorithm on price detection confidence. The price detection confidence with the Assisted Rekognition is 76.49%, whereas without the algorithm is 20.94%. The Assisted Rekognition algorithm ran on each image's nonfiltered (original) and filtered versions. Nonfiltered and filtered dataset images in each grocery group are evaluated based on price detection confidence value calculated by the Assisted Rekognition Algorithm. According to the average confidence results, the Assisted Rekognition algorithm achieved better price detection confidence on filtered images than on nonfiltered images for all grocery datasets, as shown in Figure 9. This result indicates that filtering images help the Assisted Rekognition Algorithm gain better price detection confidence.

Figure 10 demonstrates the number of times applied filters yield the best price detection confidence. On the nonfiltered images, the Assisted Rekognition algorithm gained the best price detection confidence only on two images out of 84. The Assisted Rekognition algorithm mostly achieved the best price detection confidence on the filtered images, especially on the sharpness-filtered images with Kernel1. Using the Assisted Rekognition algorithm, the best price detection confidence score is taken on the sharpness-filtered images with Kernel1 twelve times, more than those achieved with different filters.

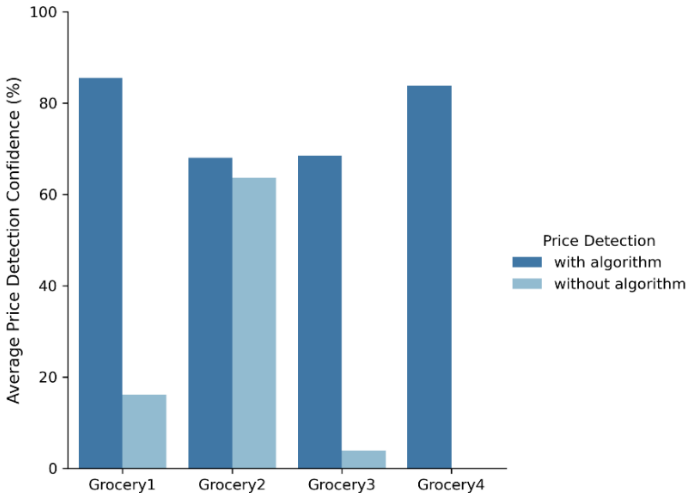


Figure 8. Average price detection confidences with and without the Assisted Rekognition algorithm

2.2. Multi-Filter Effects

The Assisted Rekognition algorithm extracted the correct price texts on 79 images out of 84 images of the FourGroceries dataset's nonfiltered images. The Assisted Rekognition algorithm could not extract the accurate price text on four images even after single filters were applied. The solution for achieving price detection on these images by the Assisted Rekognition algorithm is found to be applying multiple filters. Table 11 lists the filters applied to these images for price detection using the Assisted Rekognition Algorithm. The Assisted Rekognition algorithm extracts correct price text on FourGroceries dataset images with 100% success and with an average of 89.25% confidence by applying single or multiple filters on images. These results indicate that the ShoppingTotal mobile application will successfully output the accurate price values for shelf labels, specifically when the appropriate image filters are applied to shelf label images.

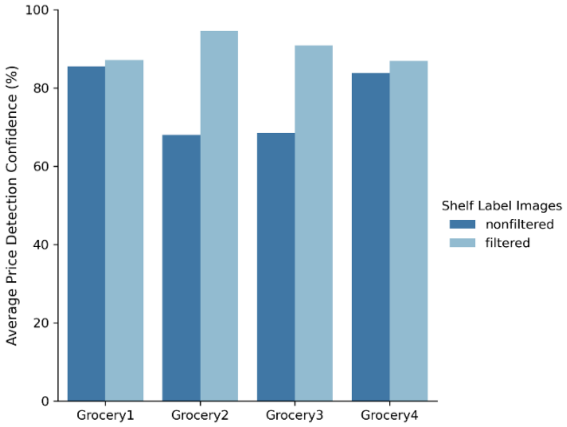


Figure 9. Average price detection confidences using the Assisted Rekognition algorithm

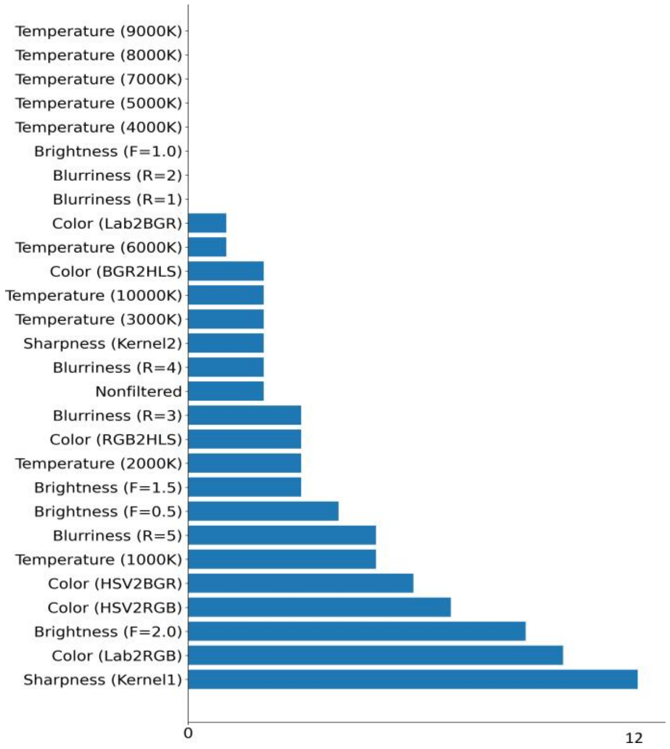


Figure 10. The number of best price detection achievements by image filtering using the Assisted Rekognition algorithm

Table 11. Multi-filter effects on the price detection success

Image Path	Price Detection Confidence (%)	Applied Filters
Grocer1/ Image_14.JPG	95.35	Sharpen (Kernel2) Color (Lab2RGB)
Grocer1/ Image_16.JPG	63.41	Brightness (Factor=2.0) Color (BGR2HLS)
Grocer4/ Image_15.JPG	92.88	Brightness (Factor=0.5) Color (Lab2BGR)
Grocer4/ Image_19.JPG	54.66	Sharpen (Kernel1) Color (Lab2RGB)

Discussion and Conclusion

This study explains the development points of a budget-monitoring mobile application, the ShoppingTotal. Technologies used to develop the ShoppingTotal mobile application can guide developers looking to build similar applications. The ShoppingTotal application extracts product information from shelf label images using the Assisted Rekognition algorithm. The Assisted Rekognition algorithm provides insights into text detection algorithms to improve accuracy and reliability.

FourGroceries dataset is presented in this paper for evaluating ShoppingTotal's price detection performance and for further studies in the area. Researchers and practitioners can use the FourGroceries Dataset in computer vision and image processing to compare and validate their approaches. The Assisted Rekognition algorithm is developed to improve the text detection confidence of Amazon Rekognition service on the FourGroceries dataset. Improved text detection performance can contribute to more reliable and efficient systems in these domains. Evaluation of image filters contributes to understanding the potential benefits and limitations of image filtering techniques in improving the accuracy of text detection algorithms. The image filtering technique can enhance the performance of the image-processing systems by selecting and optimizing the application of image filters based on requirements and image characteristics. For this study, two sharpness filters with different degrees of sharpness effects, five blurriness filters with varying radius parameters, four brightness filters with darkening or brightening image effects, ten temperature filters within the range of 1000 – 10000 Kelvin, and six color filters with varying color effects are chosen to be applied on shelf label images to evaluate the application's performance through the underlying text detection algorithm. Experiments based on these filters show that the Assisted Rekognition algorithm performs better on filtered images than on original images with 89.25% price detection confidence and extracts correct price values from shelf label images when appropriate single or multiple filters are applied to images.

Research Ethics

The author declares that the research does not have an unethical problem, and they observe the topic of research and publication ethics.

Contribution Rate of Researchers

The author read and approved the final version of the paper.

Conflict of Interest

The author declares that the study has no conflicts of interest.

Acknowledgement

Thanks to Elif Esra Eker and Aleyna Işıkdağlılar for data collection and mobile application coding.

Genişletilmiş Özet

Giriş

Alışveriş yaparken istenmeyen satın almalar nedeniyle bütçenin aşılması yaygın bir sorundur. İnsanlar genellikle almak için gittikleri ürünlerin dışında ekstra ürünler de satın alabilirler. Bu plansız harcamalar, ödeme sırasında bütçeyi aşmalarına yol açabilir. Bu sorunu çözmek için alışveriş yapanlar, alışveriş bütçelerini izlemelerine yardımcı olan bir uygulama kullanabilirler. ShoppingTotal, alışveriş sepeti toplamını raf etiketlerine göre izlemek için geliştirilmiş bir mobil uygulamadır. Alışveriş yapan kişinin onayı ile raf etiketlerinden fiyatları çıkararak alışveriş sepetine ekler. ShoppingTotal, ödeme sırasında beklenmedik fiyatlarla karşılaşmayı önlemek için raf etiketlerindeki bilgileri okur.

ShoppingTotal, Amazon Rekognition API'si ile etkileşim halindedir ve resim metinlerini algılar. Assisted Rekognition algoritması, Amazon Rekognition tarafından tespit edilen metinlerden fiyat değerlerini almak için geliştirilmiştir. Assisted Rekognition'nın performansı, 2022 yılında, dört farklı marketten toplanan raf etiketlerinden oluşan FourGroceries veri kümesi üzerinde değerlendirilmiştir. Bu veri kümesinde farklı filtrelerle işlenen görüntüler üzerinde algoritmanın performansı test edilmiştir.

Yöntem

ShoppingTotal mobil uygulaması Kotlin dilinde Android Studio üzerinde geliştirildi. Web API, market ürünlerinin raf etiketi görsellerindeki fiyat metnini Amazon Rekognition sunucusuna C# dilinde Visual Studio üzerinde geliştirilen isteklerle gönderir. Web API, Amazon Web Services (AWS) Bulutunda, EC2 üzerinde çalışır. Retrofit, web API ile Kotlin kodu arasındaki bağlantıyı sağlar. Çekilen etiket görüntüleri Amazon S3 Bucket'te saklanır ve metin tespiti için Amazon Rekognition hizmetine gönderilir. Web API, işlenen bilgileri PostgreSQL veritabanında saklar ve mobil uygulama istemcisine ürün bilgileriyle yanıt verir. Kullanıcılar, sepet içeriğini güncellediklerinde veya ürünleri sildiklerinde web API'ye istek gönderirler. Tam fiyat tespiti için Assisted Rekognition Algoritması, tespit edilen metinden olası fiyat değerlerini bütün olarak çıkarır ve güven değerini kaydeder. Kısmi fiyat tespiti durumunda ise, algoritma farklı metinlerden alınan fiyat değerlerinin ortalamasını güven değeri olarak kaydeder.

Kullanıcılar, kamera ikonuna basarak raf etiketlerini çekebilir ve web API tarafından işlenen görüntüdeki olası fiyatları görebilirler. Ürünü Sepete Ekle butonuna tıklayarak sepetlerine ekleyebilirler. Sepetim sayfasında toplam fiyatı görüntüleyebilir, ürünleri silebilir veya sayılarını güncelleyebilirler. Alışverişi Bitir butonuyla sepeti tamamlayabilir ve Eski Alışverişler sayfasından geçmiş alışveriş listelerini yönetebilirler.

ShoppingTotal uygulamasında kullanılan görüntü filtreleri, algoritmanın fiyat tespit performansını iyileştirmek amacıyla kullanılır. Bu filtreler şunlardır:

Keskinlik (Sharpness): Görüntüdeki kenarları ve detayları belirginleştirmek için kullanılır. Örneğin, Laplacian çekirdeği gibi yöntemler kullanılarak görüntü üzerinde keskinlik artırılabilir.

Bulanıklık (Blur): Görüntüdeki ayrıntıları azaltarak yumuşatma sağlar. Gauss bulanıklaştırması gibi yöntemler, algoritmanın daha doğru metin tespiti yapmasına yardımcı olabilir.

Parlaklık (Brightness): Görüntünün genel parlaklığını ayarlar. Faktör değeri artırılarak görüntü daha parlak hale getirilebilir veya azaltılarak daha koyu hale getirilebilir.

Sıcaklık (Temperature): RGB değerlerini ayarlayarak görüntüye sıcaklık efekti uygular. Bu filtre, görüntüdeki renk dengesini değiştirerek algoritmanın metin tespitini etkileyebilir.

Renk (Color): Görüntüdeki renk tonları ve doygunluğunu değiştirebilir. Örneğin, HSV veya Lab renk modelleri kullanılarak renk ayarlamaları yapılabilir.

Bu filtreler, ShoppingTotal uygulamasında kullanıcıların çektiği veya uygulama tarafından işlenen görüntüler üzerinde ön işlem yaparak, Amazon Rekognition API'sine daha doğru ve tutarlı metin tespiti için yardımcı olur. Bu sayede, algoritmanın performansı farklı görüntü koşullarında daha sağlam bir şekilde değerlendirilebilir.

Bulgular

Assisted Rekognition algoritması, FourGroceries veri kümesindeki filtrelenmemiş görsellerin 84 görselinden 79 görselden doğru fiyat metinlerini çıkarmıştır. Bu görüntüler üzerinde Assisted Rekognition algoritması ile fiyat tespiti yapmanın çözümünün birden fazla filtre uygulamak olduğu bulunmuştur. Assisted Rekognition algoritması, görüntülere tekli veya çoklu filtreler uygulayarak FourGroceries veri kümesi görüntüleri üzerinde %100 başarı ve ortalama %89.25 güvenle doğru fiyat metnini çıkarmıştır. Bu sonuçlar, özellikle raf etiketi görsellerine uygun görsel filtreleri uygulandığında, ShoppingTotal mobil uygulamasının raf etiketleri için doğru fiyat değerlerini başarılı bir şekilde çıkaracağını göstermektedir.

Tartışma ve Sonuç

Assisted Rekognition algoritması, Amazon Rekognition hizmetinin FourGroceries veri kümesindeki metin algılama güvenini artırmak için geliştirilmiştir. Geliştirilmiş metin algılama performansı, bu alanlarda daha güvenilir ve verimli sistemlere katkıda bulunabilir. Görüntü filtrelerinin değerlendirilmesi, metin algılama algoritmalarının doğruluğunun arttırılmasında, görüntü filtreleme tekniklerinin potansiyel faydalarının ve sınırlamalarının anlaşılmasına katkıda bulunabilir. Görüntü filtreleme tekniği, gereksinimlere ve görüntü özelliklerine göre görüntü filtrelerinin uygulanmasını seçip optimize ederek görüntü işleme sistemlerinin performansını artırabilir. Bu çalışma için, farklı keskinlik efektleri derecelerine sahip iki keskinlik filtresi, değişen yarıçap parametrelerine sahip beş bulanıklık filtresi, görüntüyü koyulaştıran veya parlaklaştıran dört parlaklık filtresi, 1000 – 10000 Kelvin aralığında on sıcaklık filtresi ve değişen yarıçap parametrelerine sahip altı renk filtresi kullanılmıştır. Bu filtrelere dayalı deneyler, Assisted Rekognition algoritmasının, %89.25 fiyat tespit güveniyle filtrelenmiş görseller üzerinde orijinal görsellerden daha iyi performans gösterdiğini ve görsellere uygun tekli veya çoklu filtreler uygulandığında raf etiketi görsellerinden doğru fiyat değerlerini çıkarabildiğini göstermektedir.

References/Kaynakça

Ali, R. H., Kashefi, A. K., Gorman, A. C., Walsh, J. S. P., & Linstead, E. J. (2022). Automated identification of astronauts on board the International Space Station: A case study in space archaeology. *Acta Astronautica*, *200*, 262–269.

AlWadani, R. M., & AlOtaibi, A. S. (2019). iCheck: an Android Application for Enhancing In-Store Shopping Experience Using Modern Techniques. *2019 2nd International Conference on Computer Applications \& Information Security (ICCAIS)*, 1–6.

Amazon Elastic Compute Cloud. (n.d.). Retrieved June 20, 2023, from https://aws.amazon.com/ec2/

Amazon Rekognition. (n.d.). Retrieved June 20, 2023, from https://aws.amazon.com/rekognition/

Amazon Relational Database Service. (n.d.). Retrieved June 20, 2023, from https://aws.amazon.com/rds/

Amazon Simple Storage Service. (n.d.). Retrieved June 20, 2023, from https://aws.amazon.com/s3/

Hegghammer, T. (2022). OCR with Tesseract, Amazon Textract, and Google Document AI: a benchmarking experiment. *Journal of Computational Social Science*, *5*(1), 861–882.

Kuang, X., Gao, X., Wang, L., Zhao, G., Ke, L., & Zhang, Q. (2021). A discrete cosine transform-based query efficient attack on black-box object detectors. *Information Sciences*, *546*, 596–607.

Leotta, M., Mori, F., & Ribaudo, M. (2022). Evaluating the effectiveness of automatic image captioning for web accessibility. *Universal Access in the Information Society*, 1–21.

Liu, C. Y. J., & Wilkinson, C. (2020). Image conditions for machine-based face recognition of juvenile faces. *Science \& Justice*, *60*(1), 43–52.

Out of Milk - The Grocery Shopping List App. (n.d.). Retrieved June 20, 2023, from https://www.outofmilk.com/

Scan&Shop. (n.d.). Retrieved June 20, 2023, from https://corp.lotuss.com.my/services/scan-and-shop

ShopSavvy. (n.d.). Retrieved June 20, 2023, from https://shopsavvy.com/

Siess, A. (n.d.). *RGB to color temperature*. Retrieved June 20, 2023, from https://andi-siess.de/rgb-to-color-temperature/

Yang, K., Wang, C., Sarsenbayeva, Z., Tag, B., Dingler, T., Wadley, G., & Goncalves, J. (2021). Benchmarking commercial emotion detection systems using realistic distortions of facial image datasets. *The Visual Computer*, *37*, 1447–1466.