Neural Comput & Applic (2017) 28:2859–2868
DOI 10.1007/s00521-017-2858-2

NEW TRENDS IN DATA PRE-PROCESSING METHODS FOR SIGNAL AND IMAGE CLASSIFICATION

A cloud-based recommendation service using principle component analysis–scale-invariant feature transform algorithm

Akhan Akbulut[1] · **Cagatay Catal**[1] · **Fatma Patlar Akbulut**[1]

Received: 9 November 2016 / Accepted: 9 January 2017 / Published online: 21 January 2017

Abstract Cloud computing delivers resources such as software, data, storage and servers over the Internet; its adaptable infrastructure facilitates on-demand access of computational resources. There are many benefits of cloud computing such as being scalable, paying only for consumption, improving accessibility, limiting investment costs and being environmentally friendly. Thus, many organizations have already started applying this technology to improve organizational efficiency. In this study, we developed a cloud-based book recommendation service that uses a principle component analysis–scale-invariant feature transform (PCA-SIFT) feature detector algorithm to recommend book(s) based on a user-uploaded image of a book or collection of books. The high dimensionality of the image is reduced with the help of a principle component analysis (PCA) pre-processing technique. When the mobile application user takes a picture of a book or a collection of books, the system recognizes the image(s) and recommends similar books. The computational task is performed via the cloud infrastructure. Experimental results show the PCA-SIFT-based cloud recommendation service is promising; additionally, the application responds faster when the pre-processing technique is integrated. The proposed generic cloud-based recommendation system is flexible and highly adaptable to new environments.

Keywords Feature selection · Feature extraction · Object recognition · Cloud computing · Recommendation system

1 Introduction

Modern technology supports a user by accomplishing a multitude of daily tasks. As technology matures, it becomes adept at learning from our behaviour and routines to assist with decision tasks. A simple dinner recommendation may be made by technology that records data from food delivery, restaurant and grocery receipts. Similarly, stock market recommendations may be made by an application that analyses stock price patterns. These are just a few examples of what we call recommender systems [7]. These systems have become extremely common in recent years, especially in industrial, medical and military domains.

Recommender systems can be categorized into: content-based filtering, collaborative filtering, graph-based recommendations, stereotyping, item-centric recommendations and hybrid recommendations [4]. Each approach is associated with different success ratios for a wide variety of application areas such as medical, financial, insurance and military.

In this paper, we develop, implement and validate a system that recommends books by using an image of a book or a collection of books. The system uses an image recognition method and a recommendation engine. Users simply take a photograph of a book or bookshelf, then the cloud service analyses the image, and finally it generates a list of book recommendations. The content of the photograph is categorized by an image recognition technique to determine which books the user has already read. The system then makes a recommendation based on a pre-

✉ Akhan Akbulut
a.akbulut@iku.edu.tr

Cagatay Catal
c.catal@iku.edu.tr

Fatma Patlar Akbulut
f.patlar@iku.edu.tr

[1] Istanbul Kültür University, E5 Karayolu üzeri Bakirköy, 34156 Istanbul, Turkey

existing database of book reviews that is already categorized within the system.

Our main objective is to support readers by finding books that they are most likely to enjoy using their previous reading history. This is a natural way of recommending a book and is easily delivered using a smart phone application powered by cloud computing technology. Image recognition and pre-processing are the most important components of this system. We used a PCA-SIFT algorithm to recognize and extract information from the images. This algorithm will be explained in detail in the methodology section. To generate accurate recommendations, we used a vector space model [27]. Each book review is represented by a vector. The book similarity value is determined by using a cosine similarity method which measures the distance between vectors [11]. The vector distance between two books must fall below a certain threshold to be determined similar; this is a key step in the recommendation process.

Image recognition is an important research area in image processing. The scale-invariant feature transform (SIFT) algorithm [17] is a feature detection algorithm. It is used by many researchers and practitioners in different application domains that require image retrieval and object detection. Improved versions of the SIFT algorithm include PCA-SIFT [13], SURF (Speeded up Robust Features) [3], GSIFT [20], CSIFT [1] and ASIFT [19]. Many researchers in the computer vision field [15] are modifying index-based methods to solve open problems in image retrieval [29], object recognition [5], texture classification [14], robot localization [24] and video shot retrieval [28].

[33] analysed the performance of these algorithms for accomplishing illumination change, rotation change and blur change. They rated the capability of each algorithm to perform a specific task. Each task had a unique best algorithm. Unfortunately, there was no single algorithm that could optimally perform every task. The PCA-SIFT, however, consistently had the second-best performance for each task. Thus, this algorithm was implemented in our cloud-based recommendation system.

There are two kinds of image matching algorithms, namely global and local feature-based matching algorithms [6]. Local algorithms are more stable compared to the global feature-based matching algorithms. They generally have two steps known as key point detection and description [33].

Researchers have applied PCA-SIFT algorithm to several application areas. Gao et al. [9] applied PCA-SIFT as a feature extraction algorithm and then used a support vector machine (SVM) algorithm for the classification of images. Tianyuan et al.'s paper [30] proposed a new image matching algorithm based on PCA-SIFT, correlation coefficients and RANSAC algorithms. They stated that the new hybrid algorithm is better than the other image matching algorithms in terms of precision. Shi and Yan [26] developed a scene recognition algorithm based on maximally stable colour regions (MSCR) detector and PCA-SIFT descriptors. They concluded that this approach is robust to image deformations and faster than the standard SIFT-based algorithms. Wang and Zhang [31] proposed an enhanced PCA-SIFT approach for feature extraction and applied this approach to image-based gender classification. After classifying and analysing data from a collection of three-face images, the experimental results determined the gender classification approach that provided the highest probability of classification. Kawaji et al.'s paper [12] implemented an image-based indoor positioning system as part of a digital museum application. In this system, a PCA-SIFT algorithm was used as the image matching algorithm. Shen et al. [25] developed an image stabilization algorithm by using PCA-SIFT for feature detection and an adaptive particle filter method. They proposed a new cost function called SIFT-BMSE and reported that the algorithm is efficient and accurate. Muralidharan and Chandrasekar [22] developed an object recognition algorithm based on a combination of local features and global features. They applied a Hessian–Laplace detector and PCA-SIFT to find local features while K-nearest neighbour (KNN) and SVM algorithms were used to recognize the object. Arth et al. [2] investigated a set of local feature detectors and descriptors for embedded system domains. They applied maximally stable extremal regions (MSER) and difference of Gaussian (DoG) detectors for region selection and combined them with the PCA-SIFT algorithm. The algorithms were tested on an embedded platform, and the team determined that the approach is suitable for object recognition problems. El-gayar et al. [8] analysed five feature detection methods to detect image deformations. They reported that F-SIFT had the best performance compared to SIFT and SURF. Loncomilla's paper [16] investigated object recognition methods for robotic domains and reported that SIFT and SURF had the best performance among the descriptor algorithms.

Researchers continue to focus on the development of new image matching algorithms. Wen-Huan and Qian [32] recently proposed a new image matching algorithm and reported that the algorithm yields a higher matching rate than SURF and PCA-SIFT algorithms.

The contribution of this paper is three-fold, as follows:

- A recommendation system was developed and implemented on a cloud computing system.
- We observed that the cloud-based recommendation system responds faster when a pre-processing method was applied.

- A generic architecture for a cloud-based application was proposed and validated as part of the recommendation system.

The following section explains the methodology in detail. Section 3 provides the implementation details and experimental results. Section 4 shows the conclusion and future work.

2 Methodology

In this section, we present the proposed cloud-based recommendation system in detail. Section 2.1 describes the system architecture and its components: mobile application, website, web service, feature extraction and matching module, and recommendation engine. Section 2.2 describes the SIFT and PCA-SIFT algorithms. Section 2.3 presents the recommendation approach we adopted the system.

2.1 System architecture

Our proposed system consists of seven modules, each performing a unique task. Each module, except the mobile application, is deployed in the cloud environment. The mobile application and website are the visual components of the system. The web service is used as a back-end to invoke feature extraction and recommendation engine modules. The SQL Server database and book cover collection elements feed the system with appropriate data. The entire system architecture is shown in Fig. 1.

2.1.1 *Mobile application*

An Android-based mobile application was developed to interact with the users in the system. The main aim of this mobile application is to transfer the user input (image data) to the cloud system. The process begins with the transmission of the photograph to the cloud environment. Android version 5.0 Lollipop has been selected as the mobile operating system and Android Studio IDE is used in the development phases. While the current mobile application works on Android OS, the next version will operate on both iOS and Android devices. Since the development team has extensive experience with Java technologies, we preferred to start with the Android OS.

The mobile application consists of seven interfaces. The book recognizer picture is displayed on the screen for 3 s after the application is opened. The filtering interface is then displayed on the screen. Here, the user first determines the criteria of the book (Fig. 2a); after the selection, the camera function is initialized to ensure the user takes a picture of the book(s). Next, the image is shown to the user for approval. If the user does not approve this image, the user may retake the image of the object by returning to the

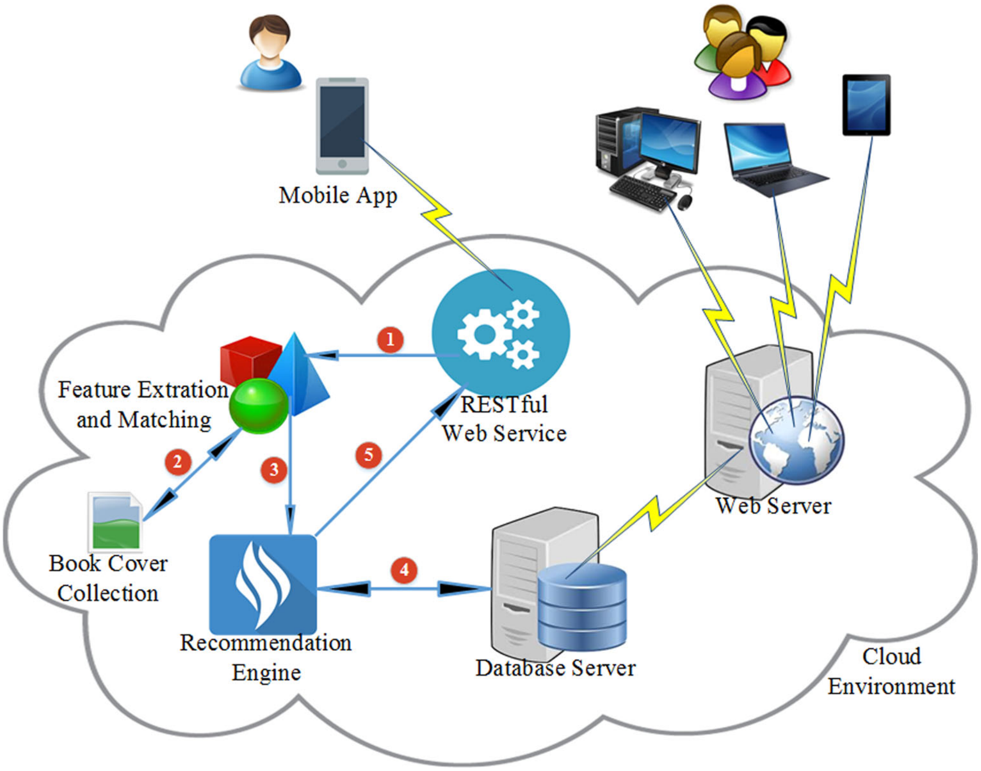
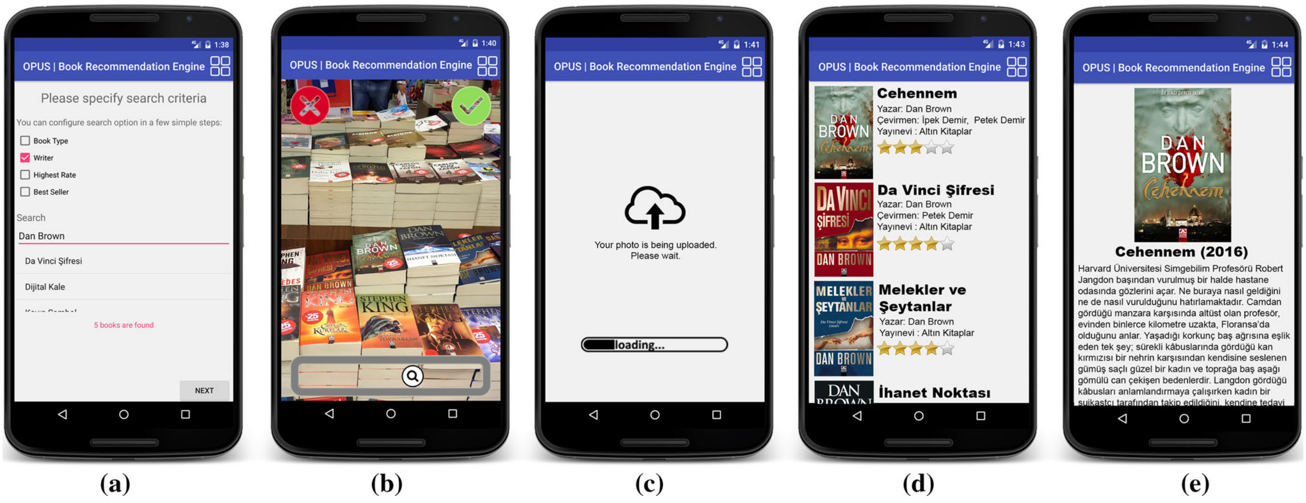


Fig. 1 Overview of entire system



OPUS | Book Recommendation Engine

Please specify search criteria

You can configure search option in a few simple steps:

□ Book Type
☑ Writer
□ Highest Rate
□ Best Seller

Search

Dan Brown

Da Vinci Şifresi

Dijital Kale

5 books are found

Your photo is being uploaded.
Please wait.

Cehennem
Yazar: Dan Brown
Çevirmen: İpek Demir, Petek Demir
Yayınevi : Altın Kitaplar
★★★☆☆

Da Vinci Şifresi
Yazar: Dan Brown
Çevirmen: Petek Demir
Yayınevi : Altın Kitaplar
★★★★☆

Melekler ve Şeytanlar
Yazar: Dan Brown
Yayınevi : Altın Kitaplar
★★★★☆

İhanet Noktası

Cehennem (2016)

Harvard Üniversitesi Simgebilim Profesörü Robert Jangdon başından vurulmuş bir halde hastane odasında gözlerini açar. Ne buraya nasıl geldiğini ne de nasıl vurulduğunu hatırlamaktadır. Camdan gördüğü manzara karşısında altüst olan profesör, evinden binlerce kilometre uzakta, Floransa'da olduğunu anlar. Yaşadığı korkunç baş ağrısına eşlik eden tek şey; sürekli kâbuslarında gördüğü kan kırmızısı bir nehrin karşısından kendisine seslenen gümüş saçlı güzel bir kadın ve toprağa baş aşağı gömülü can çekişen bedenlerdir. Langdon gördüğü kâbusları anlamlandırmaya çalışırken kadın bir suikastçı tarafından takip edildiğini, kendine tedavi

Fig. 2 Screenshots of Mobile App. **a** Filter screen, **b** photograph confirmation screen, **c** processing screen, **d** results screen and **e** details screen

first page. If the user approves the image, the image is delivered to the cloud module for processing (Fig. 2b). In the fourth screen, a process bar is displayed while the matching process is performed on the web service (Fig. 2c). At the fifth interface, the list composed per the criteria that the user selected is displayed on-screen (Fig. 2d). In the final screens, the user can view information about the book and in another screen, can see the picture of the book cover on-screen (Fig. 2e).

2.1.2 Website

The system may also be accessed via a website which is hosted by a virtual server on the cloud. The website is an alternative for users who do not want to install and use a mobile application. Users can review and read comments about books by using the website. Unlike the mobile application, website users cannot use the recommendation service because they cannot send application-specific photographs in their environment. An ASP.net platform has been preferred during the development of the front end because of its high compatibility with Microsoft-based Azure servers.

2.1.3 Web service

To ensure energy efficiency in the mobile application, we implemented a service oriented approach to realize heavy tasks such as image recognition and book recommendations. Photographs taken with the mobile application are transferred to the cloud environment via a RESTful [21] web service. This service is realized on the WCF platform of Microsoft. The main duty of this service is to create a bridge between the mobile application and cloud service. The web service acts as a coordinator in the cloud and orchestrates the main flow of the recommendation processes, namely feature extraction, cover matching and sending recommendation results. After carefully considering performance issues, a REST form was preferred over a SOAP implementation.

2.1.4 Feature extraction and matching module

The most important phase of the recommendation process is to extract features from the photograph to determine the name of the book(s) the user has read. The PCA-SIFT algorithm, which classifies the images, is performed as a background process using a worker role of the Azure platform. PCA-SIFT is an enhanced version of the SIFT feature extraction algorithm which is superior to its counterparts SURF, GLOH and BICE. This feature extraction method is described in detail in Sect. 2.2.

2.1.5 Book cover collection

The PCA-SIFT algorithm compares two images, with extracted features, to find matching book covers. The first parameter is the photograph that is transmitted to the cloud environment from the mobile application. The image is compared with a collection of images whose features are stored in a database. The structure of the book cover collection is explained in the Implementation and Results section.

2.1.6 Recommendation engine

The recommendation engine processes have been implemented as worker roles in an Azure Cloud Computing platform. Collaborative filtering techniques were used to implement the recommendation engine after several problems with content-based filtering approaches were encountered. For instance, certain book properties that were not machine-recognizable were capable of being used as features. Also, processing book content is time-consuming and expensive. In addition to these drawbacks, content-based recommendation systems do not use personal recommendations to feed the system. Therefore, a recommendation engine based on collaborative filtering techniques was identified as the best solution.

2.1.7 Database server

SQL Server 2014, a relational database management system, was selected to store the system data. User and book details, reviews and comments are the main variables of our database. By using the cloud system's elasticity feature, database load balance is managed dynamically.

2.2 Feature extraction based on PCA-SIFT

Since there are many feature extraction methods that find point correspondences between two images in the literature, we decided to use PCA-SIFT algorithm for its high performance. We describe the SIFT algorithm in detail next. Then we demonstrate the differences between SIFT and PCA-SIFT. We will highlight the modification steps for our system when appropriate.

Scale-invariant feature transform (SIFT) The SIFT [18] algorithm is a feature detection method that is used to extract local features from objects, regardless of scale. SIFT features are found using orientation histograms from locations around a key point in scale space, which has 128 dimensional (orthogonal) vectors. The most important feature of this method is that the input data are transformed to data that are invariant to scale and rotation, noise and robust to perspective changes. As extracted feature vectors are independent of scale and rotation, a distance measure between vectors can be used to match objects and environments.

SIFT contains four main steps [18]: scale space extrema detection, key point localization, orientation assignment and key point descriptors. In the first step, the algorithm searches over multiple scales and images to identify locations and scales which can be repeatedly matched under different views of the same image. This task is performed by organizing data as a Gaussian pyramid and searching for key points in a series of difference of Gaussian (DoG) images. Once a key point candidate is found, we compare it to nearby data points to determine maxima and minima of DoG images in scale space. Next, we select key points with minimum stable distance measures. Each key point is measured against its eight neighbours in the original image and nine neighbours in each of the scaled images. Following this process, a Hessian matrix is used to eliminate divergent edge responses. The third step to identify dominant orientations and create histogram of local gradient directions by assigning canonical orientation. This step aims to remove all effects of rotation and scale. The final step determines a descriptor of a local image feature for each key point. The local dominant gradient angle is calculated and used as a reference orientation. The local gradient distribution is normalized per the reference direction.

SIFT using principal component analysis (PCA-SIFT) PCA-SIFT [13] is the improved version of SIFT which intends to decrease high dimensionality in SIFT and eliminate the computational cost for image indexing and retrieval by using principal component analysis. The modification replaces the histogram step with a step that normalizes gradient patches; this will offer the opportunity to work with relatively smaller feature vectors compared to SIFT even if the same matching algorithm is used. This modification of the SIFT algorithm starts at the fourth step of application. The steps of entire process are as follows:

In the first step, we determine extrema points to normalize the feature vectors to unit vectors. The difference of Gaussian transform is then convolved with the 2D image to transform the image data to scale space image data. This step is described mathematically as follows:

$$\begin{aligned} D(x,y,\sigma) &= (G(x,y,k\sigma) - G(x,y,\sigma)) * I(x,y) \\ &= L(x,y,k\sigma) - L(x,y,\sigma) \end{aligned} \tag{1}$$

$I(x, y)$: image

$L(x,y,k\sigma)$: scale space function

where $G(x,y,\sigma)$ is a scale variable Gaussian Function:

$$G(x,y,\sigma) = \frac{1}{2\pi\sigma^2} e^{\frac{-(x^2+y^2)}{2\sigma^2}} \tag{2}$$

And σ is used to determine image smoothness.

Various Gaussian scale factors $G(x,y,\sigma)$ of an image $I(x, y)$ are shown in the following figure.

Figure 3 illustrates Gaussian pyramid.

This is image is achieved by applying Gaussian filters, each having different sigmas, to the original image data. As you can see, each image is progressively blurred and the difference between sequential blurring factors generates an Octave of the pyramid.

The following operation is a key point localization step which is used to search each pixel in the DoG map to find the extrema. To do this, we ignore the points with low contrast or blurred edges.

Figure 4 illustrates key point localization.

In the third step, we assign an index to each orientation. Next, we detect the searchable area for each image point and create a weight matrix. We then calculate each regions pixel gradient magnitude $s(x, y)$ and orientation $o(x, y)$.

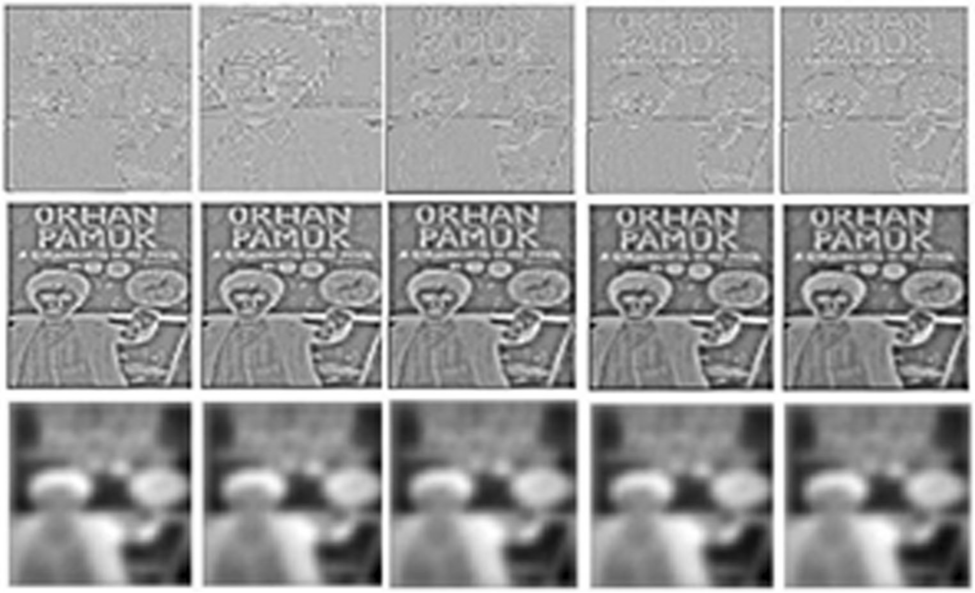


Fig. 3 Camera distances different from the original. There are 4 octaves and 5 images

The next step is to find the histogram bar and set a maximum threshold of 80%.

$$s(x,y) = \sqrt{(L(x+1,y) - L(x-1,y)^2) + (L(x,y+1) - L(x,y-1)^2} \quad (3)$$

$$o(x,y) = \tan^{-1}\frac{L(x,y+1) - L(x,y-1)}{L(x+1,y) - L(x-1,y)} \quad (4)$$

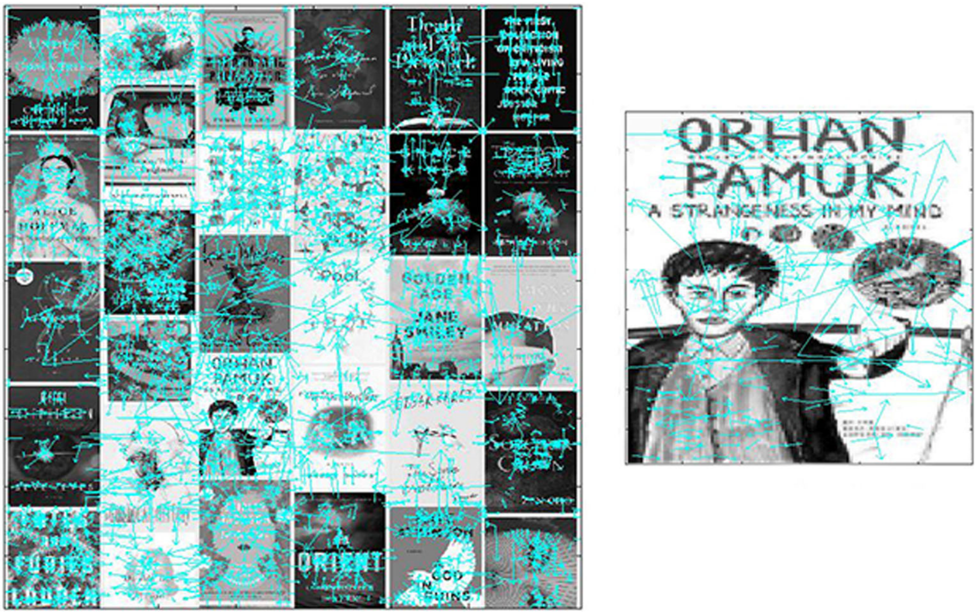
Notice that the first three steps are the same as SIFT. In the fourth step, we use PCA instead of weighted histograms and concatenate the gradients into a single vector.

$$A_{KXN}I_{NX1} = I'_{KX1} \quad (5)$$

Next, we extract a 41 × 41-pixel image patch around each key point and compute the horizontal and vertical gradients which yields a gradient field ($39 \times 39 \times 2 = 3042$). We then find the covariance matrix of A_{KXN}. As a final step, eigenvectors and eigenvalues of the resulting covariance matrix are found.

Euclidean distance is used to determine if the pairs of key point descriptors in different images are the same or if the distance falls below a predetermined threshold as shown in Figs. 5 and 6.

Fig. 4 This figure shows key point selection



2.3 Recommendation approach

Our recommendation component specifically uses an item-based collaborative filtering algorithm. We analyse the items, which are rated by the user, compute their similarity and output a subset of the most similar items [23]. Once the similar books subset is found, the prediction step ranks each book based on the weighted average of the users rating of the original bookshelf item. Several similarity computation methods exist: correlation-based similarity, cosine similarity and adjusted-cosine similarity. In our implementation, rows were used to index users and columns indexed items. Similarities were computed using cosine similarities between item vectors. Equation 6 shows the cosine similarity formula. Two items, in vector form, were compared by calculating the cosine angle between the two vectors. vectors. The prediction step is also important for recommendation systems, and we applied a weighted average approach as it was practical.

$$\text{cosine-similarity}(A, B) = \frac{\langle A, B \rangle}{||A|| \cdot ||B||} \tag{6}$$

3 Implementation and results

The most important part of this cloud-based system was to find the book covers in the photograph taken by the mobile application. As we explained the internal details of our feature extraction and matching module in Sect. 2, this part will focus on the execution parameters and results.

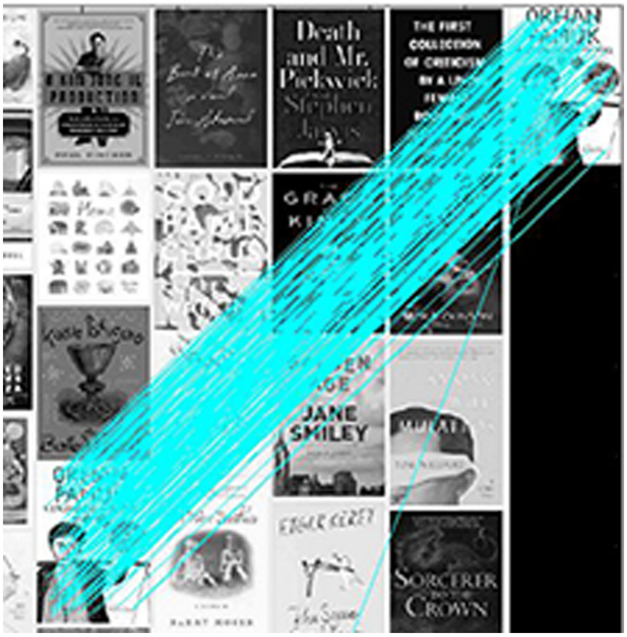


Fig. 5 This figure shows the matching result

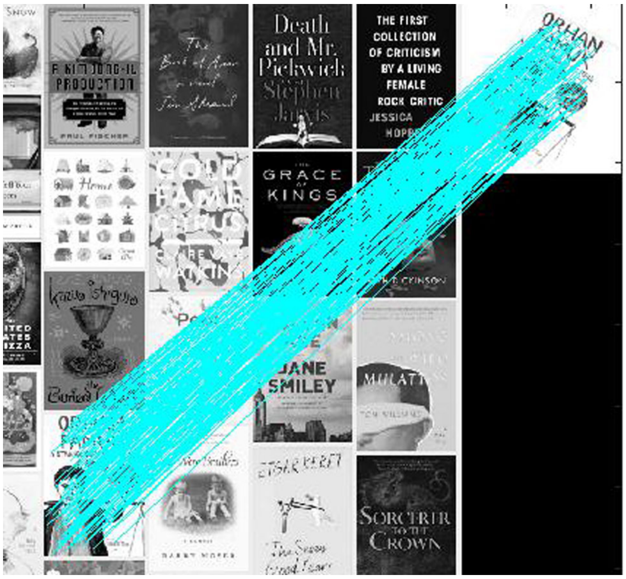


Fig. 6 This figure shows the matching result of rotated image

Our alpha version compared the photograph taken with the mobile application with a single book cover by using a loop which compares the input item to each record in the database. For a single image comparison, it took approximately 5 s with PCA-SIFT algorithm and 7 s with traditional SIFT algorithm. During the benchmarking tests, we used a computer which has an Intel Core i5 6400 2.70 GHz CPU and 8 GB RAM. By using this configuration, the service subroutine compared all the 3207 covers in approximately 4.5 h using the PCA-SIFT algorithm and 6.1 h using the SIFT algorithm. To evaluate the real performance, we deployed this service to two different virtual computers in the cloud infrastructure. The first virtual server is an instance of F8 model with 8 core CPUs and 16 GB memory, and the second one is a H16r instance with 16 core CPUs and 112 GB memory. While the F8-based computer performed the entire comparison less than 22 min, H16r did the same job within 8 min. The test results for this alpha version are provided in Table 1.

Since the alpha version had a 10-min run-time average, we determined the service period was unacceptable. We decided to optimize the recommendation service by modifying the configuration parameters. We observed that the high dimensionality of images during the comparisons has a negative effect on the computation time. In the first optimization we tested, we reduced the photograph size to 1/3 (1000 pixel*800 pixel) the size of the original and defined a standard height (200 pixel) to all the covers. Then, we changed the comparison logic from one image to 5 × 5 catalogues. In the previous version, the service iterates the comparison of photograph to covers one by one. By using catalogue images which show 25 cover at the same time, we reduced the execution time significantly.

Table 1 Average running results of alpha version

# of covers	Development computer		Cloud computer F8		Cloud computer H16r	
	SIFT	PCA-SIFT	SIFT	PCA-SIFT	SIFT	PCA-SIFT
1	7 s	5 s	30.9 ms	23.3 ms	11.6 ms	9.2 ms
3207	6.1 h	4.5 h	32.7 min	21.8 min	10 min	8 min

With the help of catalogues, the entire search was completed with 129 iterations instead of 3207 iterations. The comparison performed with PCA-SIFT using catalogue images is displayed in Fig. 7.

By changing the single cover image to the catalogue which holds 25 covers, we dramatically improved the system performance by 90%. This change reduced the computation time of the catalogue matching step to less than 8 s on the development computer using the PCA-SIFT algorithm. Our system's improved version results are shown in Table 2.

As seen in Table 2, we managed to optimize the service time to a reasonable execution period. When we add the upload time of the photograph to the data processing time, the total recommendation process is completed in 30 s if suitable images, in terms of illumination, blur and rotation are uploaded. In the field test, we took some blurry photographs by vibrating the smart phone during the shoot. Blurred photographs are not recognized by the system, but PCA-SIFT finds 17% more key points than SIFT. For rotated objects, such as horizontally placed books in the shelf, SIFT yields a better performance than PCA-SIFT. Considering the performance issues, PCA-SIFT has been implemented as the main algorithm for the feature selection and matching modules. By filtering the search criteria from the mobile application's interface, PCA-SIFT enabled service time to be completed in approximately 10 s.

The limitation of our cloud-based recommendation system is related to the underlying object recognition algorithm called PCA-SIFT. In the comparison of object recognition algorithms [10], PCA-SIFT is known to perform well under illumination and rotation changes; however, PCA-SIFT is not the best choice when encountering scale and blur changes. Therefore, we recommend that the distance between the mobile application and the object must be between 0.5 and 3 metres. The mobile device should also not be vibrated during the photograph shoot due to resulting blur effects.

The activity diagram of Fig. 8 depicts the entire flow of process between different platforms.

4 Conclusion and future work

Object recognition is an important research area which helps to identify the objects in a video or image in the computer vision. In this study, we developed a cloud-based recommendation system by using a PCA-SIFT feature detector algorithm which applies principle component analysis (PCA) to reduce the SIFT algorithms high dimensionality. The service was implemented using this additional pre-processing step. We observed that this cloud-based recommendation system responds faster than traditional approaches which do not apply pre-processing methods. We developed and validated this generic cloud-

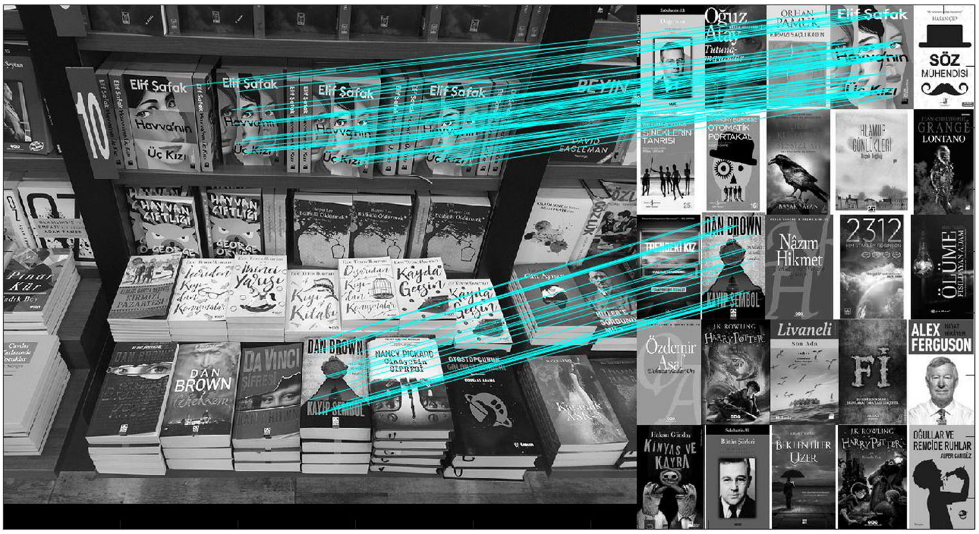


Fig. 7 PCA-SIFT algorithm using 5 × 5 catalogue

Table 2 Average execution results of final version

# of covers	Development computer		Cloud computer F8		Cloud computer H16r	
	SIFT	PCA-SIFT	SIFT	PCA-SIFT	SIFT	PCA-SIFT
25 (5 × 5)	11 s	8 s	Imponderable values			
3207	24 min	16 min	95 s	1 min	30.4 s	21 s

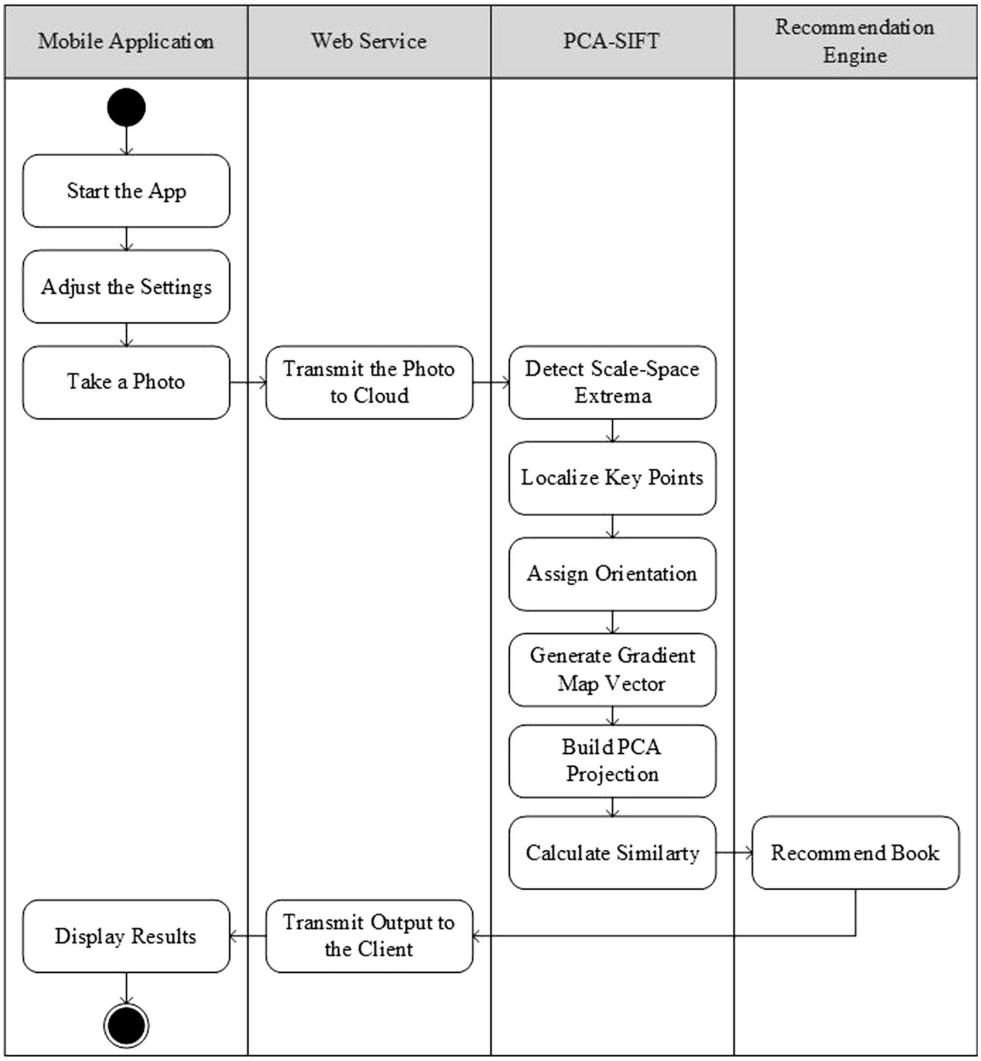


Fig. 8 Activity diagram for the cloud-based recommender system

based software architecture for an application that recommended books based on previous reading history. Since all the calculations are performed on a cloud server, the system performance is superior compared to standalone mobile applications which do not utilize web service technology. We used several tools in this project, namely Visual Studio for web service development, Eclipse for Android development, Visual Paradigm for system design and Photoshop for application user interface design.

The system is highly adaptable and flexible to the new environments and therefore, new versions will be able to address new domains such as food recommendations. In addition to the adaptability feature of this system, the usability is amazing. The end users can receive a recommendation from the system within a few seconds. This application is likely to satisfy the high expectations from mobile application users due to its run time. To demonstrate the systems effectiveness in terms of usability, we will perform usability evaluation tests and measure the usability as part of our future work. Since ISO/IEC 9126-4 Metrics standard recommends the effectiveness, efficiency and satisfaction as usability metrics, we will quantify and evaluate the usability of the implemented system. Measurement of these metrics is not a trivial task and therefore, we will focus on this challenging measurement problem as future work well. For example, satisfaction questionnaires will be used to measure the user satisfaction. Additional future work will integrate the system with social media platforms such as Twitter and Facebook to share book recommendations with friends. The other research directions include the integration of new feature detectors to the system and performing the stress testing of the system under heavy loads. This prototype system is promising and is expected to receive high user-satisfaction results.

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Abdel-Hakim AE, Farag AA (2006) CSIFT: a SIFT descriptor with color invariant characteristics. In: 2006 IEEE computer society conference on computer vision and pattern recognition (CVPR'06), vol 2, pp 1978–1983. IEEE
2. Arth C, Leistner C, Bischof H (2007) Robust local features and their application in self-calibration and object recognition on embedded systems. In: 2007 IEEE conference on computer vision and pattern recognition, pp 1–8. IEEE
3. Bay H, Ess A, Tuytelaars T, Van Gool L (2008) Speeded-up robust features (SURF). Comput Vis Image Underst 110(3):346–359
4. Beel J, Gipp B, Langer S, Breitinger C (2016) Research-paper recommender systems: a literature survey. Int J Digit Libr 17(4):305–338
5. Berg AC, Berg TL, Malik J (2005) Shape matching and object recognition using low distortion correspondences. In: 2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05), vol 1, pp 26–33. IEEE
6. Birinci M, Diaz-de Maria F, Abdollahian G, Delp EJ, Gabbouj M (2011) Neighborhood matching for object recognition algorithms based on local image features. In: Digital signal processing workshop and IEEE signal processing education workshop (DSP/SPE), 2011 IEEE, pp 157–162. IEEE
7. Bobadilla J, Ortega F, Hernando A, Gutierrez A (2013) Recommender systems survey. Knowl Based Syst 46:109–132
8. El-gayar M, Soliman H et al (2013) A comparative study of image low level feature extraction algorithms. Egypt Inform J 14(2):175–181
9. Gao H, Liu C, Yu Y, Li B (2014) Traffic signs recognition based on PCA-SIFT. In: Intelligent control and automation (WCICA), 2014 11th world congress on, pp 5070–5076. IEEE
10. Juan L, Gwun O (2009) A comparison of SIFT, PCA-SIFT and SURF. Int J Image Process 3(4):143–152
11. Kaur S, Aggarwal D (2013) Image content based retrieval system using cosine similarity for skin disease images. Adv Comput Sci Int J 2(4):89–95
12. Kawaji H, Hatada K, Yamasaki T, Aizawa K (2010) An image-based indoor positioning for digital museum applications. In: Virtual systems and multimedia (VSMM), 2010 16th international conference on, pp 105–111. IEEE
13. Ke Y, Sukthankar R (2004) PCA-SIFT: A more distinctive representation for local image descriptors. In: Computer vision and pattern recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE computer society conference on, vol 2, pp II-506. IEEE
14. Lazebnik S, Schmid C, Ponce J (2005) A sparse texture representation using local affine regions. IEEE Trans Pattern Anal Mach Intell 27(8):1265–1278
15. Li J, Allinson NM (2008) A comprehensive review of current local features for computer vision. Neurocomputing 71(10):1771–1787
16. Loncomilla P, Ruiz-del-Solar J, Martínez L (2016) Object recognition using local invariant features for robotic applications: a survey. Pattern Recognit 60:499–514. doi:10.1016/j.patcog.2016.05.021
17. Lowe DG (1999) Object recognition from local scale-invariant features. In: Computer vision, 1999. The proceedings of the seventh IEEE international conference on, vol 2, pp 1150–1157. IEEE
18. Lowe DG (2004) Distinctive image features from scale-invariant keypoints. Int J Comput Vis 60(2):91–110
19. Morel JM, Yu G (2009) ASIFT: a new framework for fully affine invariant image comparison. SIAM J Imaging Sci 2(2):438–469
20. Mortensen EN, Deng H, Shapiro L (2005) A sift descriptor with global context. In: 2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05), vol 1, pp 184–190. IEEE
21. Mumbaikar S, Padiya P (2013) Web services based on soap and rest principles. Int J Sci Res Publ 3(5)
22. Muralidharan R, Chandrasekar C (2012) Combining local and global feature for object recognition using SVM-KNN. In: Pattern recognition, informatics and medical engineering (PRIME), 2012 international conference on, pp 1–7. IEEE
23. Sarwar B, Karypis G, Konstan J, Riedl J (2001) Item-based collaborative filtering recommendation algorithms. In: Proceedings of the 10th international conference on World Wide Web, pp 285–295. ACM
24. Se S, Lowe D, Little J (2001) Vision-based mobile robot localization and mapping using scale-invariant features. In: Robotics and automation, 2001. Proceedings 2001 ICRA. IEEE international conference on, vol 2, pp 2051–2058. IEEE
25. Shen Y, Guturu P, Damarla T, Buckles BP, Namuduri KR (2009) Video stabilization using principal component analysis and scale invariant feature transform in particle filter framework. IEEE Trans Consum Electron 55(3):1714–1721
26. Shi DC, Yan GQ (2010) Combining MSCR detector and PCA-SIFT descriptor for scene recognition. In: Advanced computer control (ICACC), 2010 2nd international conference on, vol 2, pp 136–141. IEEE
27. Singh VK, Singh VK (2015) Vector space model: an information retrieval system. Int J Adv Eng Res 141:143
28. Sivic J, Schaffalitzky F, Zisserman A (2006) Object level grouping for video shots. Int J Comput Vis 67(2):189–210
29. Tao D, Tang X, Li X, Rui Y (2006) Direct kernel biased discriminant analysis: a new content-based image retrieval relevance feedback algorithm. IEEE Trans Multimed 8(4):716–727
30. Tianyuan T, Zhiwei K, Jin L, Xin H (2015) Small celestial body image feature matching method based on PCA-SIFT. In: Control conference (CCC), 2015 34th Chinese, pp 4629–4634. IEEE
31. Wang Y, Zhang N (2009) Gender classification based on enhanced PCA-SIFT facial features. In: 2009 first international conference on information science and engineering, pp 1262–1265. IEEE
32. Wen-Huan W, Qian Z (2014) A novel image matching algorithm using local description. In: Wavelet active media technology and information processing (ICCWAMTIP), 2014 11th international computer conference on, pp 257–260. IEEE
33. Wu J, Cui Z, Sheng VS, Zhao P, Su D, Gong S (2013) A comparative study of sift and its variants. Meas Sci Rev 13(3):122–131