Journal of
Data science and Modern Techniques

Research Article Open Access

A Study on Product Categorization Using Machine Learning Clustering of Image Data Using Convolutional Processing

Yoshikazu Sakamaki*

Seigakuin University, Saitama Pref. Japan

***Corresponding author:** Yoshikazu Sakamaki, Seigakuin University, Saitama Pref. Japan, Tel: 819092036999, Email: megacity@f3.dion.ne.jp

Received Date: April 20, 2024 **Accepted Date:** May 20, 2024 **Published Date:** May 23, 2024

Citation: Yoshikazu Sakamaki (2024) A Study on Product Categorization Using Machine Learning Clustering of Image Data Using Convolutional Processing. J Data Sci Mod Tech 1:1-18.

Abstract

When classifying products using cluster analysis, a general approach is to quantify attributessuch as the price, color, and size of products before analysis. However, products in online shops are often introduced using images, and when analyzing products, the necessary attribute information mustbe collected again as data, which results in a larger workload.

In the present study, we use the widely utilized convolutional processing and extract the attribute information of products such as the form of a feature vector. We then propose a method for classifying products by similar characteristics using the extracted feature vector. By applying the proposed method to actual image data on fashion products, we show that products can be classified onthe basis of only the information obtained from image data via a verification experiment.

Keywords: Machine Learning, Convolutional Processing, K-means method, Clustering

Introduction

In implementing effective sales strategies in a modern market where a multitude of products are distributed, it is often the case that customers or products are categorized into small groups with similar characteristics and the same promotion is conducted for customers or products in the groups. In many cases, classification is performed on the basis of product attributes or past purchase history and cluster analysis is conventionally used.

Looking at previous studies, when classifying products using cluster analysis, it is common to quantify attributes such as product price, color, and size to conduct the analysis. Because of the expansion of e-commerce, the number of stores that introduce products by making full use of image data instead of product attributes with letters and numerical values is increasing on the internet. When introducing products using images, it is not necessary to give detailed explanations regarding product attributes using characters and numerical values so product introductions using product images are the mainstream in online shops.

When introducing products using images in an online store, detailed attribute information regarding the target product is often not prepared as characters or numerical values. In such a case, the attribute information required for classifying the products must be collected again, which causes a large workload. Additionally, an online shop handles a large number of products, and digitizing attribute information for each product results in enormous cost and work time for marketers. If attribute information can be extracted from the image data, it will become possible to classify the products only with the information obtained from the image data. This not only reduces the workload of marketers but can also lead to a more efficient recommendation of products to consumers, providing a great advantage for companies that operate online shops.

Thus, in this study, we attempt to use convolutional processing, which is widely used today in machine learning to extract the attribute information of online products in the form of a feature vector. Then, we propose a method for classifying products into those with similar characteristics using feature vectors. By applying the proposed method to actual data gathered from fashion products, we show that products can be classified on the basis of only the information obtained from image data through a verification experiment.

Outline of Previous Studies

In various industries, there is an accelerating movement to analyze a large amount of data accumulated in a company and to make management decisions on the basis of the knowledge obtained from data analysis. This movement is known as data-driven management, and decision making based on data analysis can be seen in various marketing situations ranging from manufacturing sites to personnel planning.

Looking at examples of data utilization in the marketing field, it is a common practice to classify objects such as products, stores, and customers into groups with similar characteristics on the basis of their respective attributes. For example, on the basis of the analysis results, marketers can recommend products to customers with similar characteristics or similar to those viewed in the past on the online shop. The strategy is accomplished on the basis of the idea of classifying the targets into similar groups and then implementing the same marketing strategy for products or customers belonging to the same group.

Since the mid-1950s, various algorithms such as discriminant analysis, cluster analysis, and self- organizing maps have been studied as methods for classifying objects into those with similar characteristics [1].

Anderson [2] proposed a method of classifying data into two classes by discriminant analysis. Studies on the classification of data by cluster analysis can be traced back to Fisher's study [3]. Fisher proposed a method to classify data after grouping it into data with close values when the data is one variable. The algorithm proposed by Fisher calculates the weighted squared error from the mean value of the data in the group and forms the group so that the total of the squared errors is minimized. Cox [4], proposed an algorithm for classifying data for special cases where the data are normally distributed. Ward [5], proposed an algorithm for hierarchically combining data by first grouping the data closest to each other and then repeating the combination of the groups for multidimensional data. The algorithm proposed by Ward came to be called the Ward method and is still widely used as a typical algorithm in cluster analysis.

After that, MacQueen [6], proposed an algorithm called the K-means method which is a multidimensional extension of the algorithm proposed by Fisher. The K-means method is a simple algorithm, relatively easy to understand, and can be applied to a large amount of data. Thus, it is currently used in various fields as the main classification algorithm in machine learning.

Frank and Green [7] reported one of the earliest studies that applied the K-means method to marketing. Frank *et al.* conducted a study to classify television programs into those with similar characteristics using the Euclidean distance based on the attribute information of the program. Additionally, George and Roger [8] conducted a study to classify stores into those with similar characteristics using the Euclidean distance based on the attribute values extracted using principal component analysis. Since these studies, the K-means method has become widely used as an algorithm for classifying objects.

Since then, cluster analysis by the Ward method and the K-means method has been used in various decision making contexts.

In stores that handle many products and customers, it is essential to classify products and customers into groups with similar characteristics to efficiently promote sales strategies.

Looking at the decision making paradigm in marketing, since previously mentioned studies, data analysis for decision making is conducted after quantifying the attribute information of products, stores, and customers. Looking at the world of e-commerce, a huge number of products are sold in online shops. Looking around the web pages of e-commerce companies, it is common for only product images to be displayed and that become the only information obtained from the image data.

In previous studies, classification of products has been done mostly on the basis of quantified product attributes but considering the current state of e-commerce, it may be possible to streamline store management if products can be classified using only information obtained from image data. Thus, in this study, we attempt to propose a method for classifying products into groups with similar characteristics by performing cluster analysis using only the information obtained from the images.

Proposed Method in this Study

As data utilization progresses in various fields, research on machine learning has been increasingly conducted in recent years. Deep learning is one of the typical algorithms used in machine learning and learning algorithms using neural networks are often collectively called deep learning.

In recent years, algorithms for learning by neural networks using image data as an input layer have also been widely studied [9]. Most of the algorithms perform convolutional processing for image data as preprocessing and then perform learning using a neural network that incorporates feature vectors extracted from convolutional processing as an input layer. This algorithm is generally called a convolutional neural network (CNN) and is being applied in various fields to achieve significant results [10,11].

However, most of the previous studies on the classification of image data based on a CNN are of the type that predicts the classification result under the condition that the teacher data are given. So far, few studies have attempted to classify image data under the condition that there are no teacher data.

In this section, we propose an algorithm that enables the classification of image data without the provision of teacher data by extracting feature vectors from image data and classifying feature vectors into groups with similar features by cluster analysis.

Outline of the proposed method

The outline of the proposed method in this study is shown below. The symbols used in this section are defined as follows:

k	Image number ($k = 1,2,\ldots.,K$)
i	Horizontal pixel coordinates ($i = 1,2,\ldots.,I$)
j	Vertical pixel coordinates ($j = 1,2,\ldots.,J$)

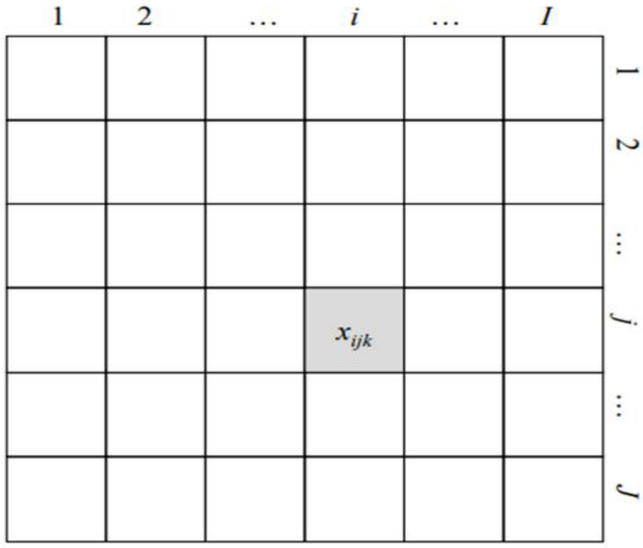


Figure 1: Image of pixel coordinate

Moreover, the pixel value of the pixel coordinate [*i j*] in the image data *k* is described as $xijk$.

These image data are classified according to the following steps:

Step.1: Implementation of convolutional processing

Convolutional processing is performed by passing the image data through filters. The filters used here are called the kernel filter and pooling filter.

In this study, we define the number of the kernel filter as *h* (*h* = 1,2,….,*H*) and the weight assigned to the filter coordinates (*s*, *t*) as $wsth$(*s*=1,2,…,*S*、*t*=1,2,….,*T*).

However, since a square matrix is generally used for the filter, S = T. Additionally, the coordinates of the data generated by performing the convolutional processing using the filter *h* on the image number *k* are $u[m,n]h$(*m* = 1,2,…,*M*、*n* = 1,2,….,*N*).

When the slide width of the filter is described as *b*, the value of the coordinates $u[m,n]h$ is formulated by equation (1):

$$u[m,n]_h = \sum\nolimits_{t=1}^{T} \sum\nolimits_{S=1}^{S} w_{s,t,h} x_{b*(m-1)+s,b*(n-1)+t}. \tag{1}$$

And maximum value of *M* and *N* is formulated by equations (2) and (3):

$$M = floor\left(\frac{I-S}{b}\right)+1, \tag{2}$$

$$N = floor\left(\frac{J-S}{b}\right)+1. \tag{3}$$

However, floor (*x*) is a function that truncates the decimal point of *x*.

Step.2: Data reduction by max pooling

Next, by passing the data of the convolutional layer through a filter, the data size is reduced while leaving only essential information. The filter used here is called the pooling filter, and the data group extracted by passing through the filter is called the pooling layer.

We describe the coordinates of the pooling filter used in this study as (*p*, *q*) (*p* = 1,2,…, *P*、*q* = 1,2,…,*Q*). However, since a square matrix is generally used for the filter, P = Q.

We also describe the coordinates of the pooling layer

$$E = floor\left(\frac{M-P}{c*P}\right)+1. \tag{4}$$

$$G = floor\left(\frac{N-P}{c*P}\right)+1. \tag{5}$$

generated by pooling processing as $v[e, g]h$ (*e* = 1,2,…,*E*、*g* = 1,2,…,*G*). When the slide width of the pooling filter is given by

$$V[e,g]_h = \max_{p\,\in\,1,2,\dots,P\ ;\ q\,\in\,1,2,\dots,Q} (u[(e-1)*P+p,(g-1)*Q+q]_h) \tag{6}$$

c, the maximum value of *E* and *G* is formulated by equations (4) and (5):

Then, the data with the largest value are selected from the area surrounded by the filter.

$$A_k = [z_{1k}, z_{2k},, \dots z_{AK}] \tag{7}$$

$A = E \times G \times H.$

Step.3: Converting data in pooling layer to vector format

Furthermore, the data group in the pooling layer in the matrix format is converted into the vectorformat. This conversion is called Flatten. We describe the vector generated by Flatten as **A***k*. However,

We call **A***k* the feature vector of image data *k*.

Figure 2 summarizes the flow of feature vector extraction via convolutional processing.

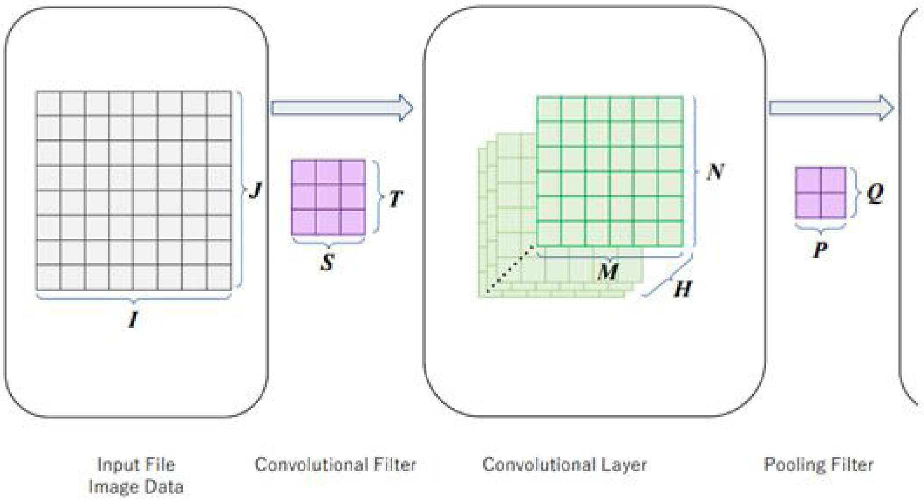


Figure 2: Flow of convolutional processing

Step.4: Classification using the K-means method

We classify the image data by the K-means method using the feature vector $\mathbf{A}_k$ extracted in Step3 as a variable. Classification is performed by the following procedures:

Procedure 1 Determine the number of clusters C to generate.

Procedure 2 Set C random points in the data, and use them as the initial value of the centroids.

Procedure 3 Classify data into groups by assigning all data to the closest centroid. Assign all the points to the closest cluster centroid.

Procedure 4 Recompute the centroid within newly created groups.

Procedure 5 Repeat Procedures 3 and 4 until the position of the centroids is converged.

Procedure 6 When the position of the centroids converges, the within-cluster's sum of squared error (SSE) is calculated. When the number of clusters is set to C, SSE is formulated byequation (8):

$$SSE_c = \sum_{c=1}^{c} \left(\sum_{k=1,k\in c}^{k_c} \sqrt{\sum_{a=1}^{A} (z_{ak} - \bar{z}_a)^2} \right) \tag{8}$$

Finally, the optimum number of clusters is determined on the basis of equation (8). The number of clusters is generally determined by the following procedure. That is, the number of clusters C is increased to 2, 3, 4, …, and the relationship between the number of clusters and the SSE is plotted on agraph to create a line graph. This line graph is called an elbow graph. Generally, SSE decreases monotonically as the number of clusters increases, but the decreasing tendency becomes gradual after a certain number of clusters. The optimum number of clusters is decided by the number of clusters thatis the boundary where the decreasing tendency changes.

Comparison method

In this study, as a comparison method with the proposed method, we attempt to extract pixel values from the image data before convolutional processing and perform clustering via the K-means method on the basis of the pixel values. Then, the results obtained from this process will be used as a comparison method in this study.

That is, on the basis of the pixel value x_{ijk} in image data k, we generate the vector $\mathbf{A}_k$ by equations (9) and (10):

Then, on the basis of $\mathbf{A}_k$, we classify image data via the K-means method according to the procedure shown in Section 3.1.

$$a = (j-1)I + i, \tag{9}$$

$$z_{ak} = x_{ijk} \tag{10}$$

Verification experiment using actual data

In this section, by applying the actual data to the proposed method and the comparison method shownin Section 3, we will show that classification based on the feature vector obtained by the convolutionalprocessing is possible to classify image data with higher accuracy than clustering based on pixel valuesobtained from original image data.

Data outline used in the verification experiment

Various datasets are prepared in a database called MNIST provided by the National Institute of Standards and Technology, which was created for introduction to machine learning. One of them is a dataset called Fashion-MNIST, which is a collection of images related to fashion products. In this study, we will conduct a verification experiment using Fashion-MNIST.

Overview of fashion-MNIST

The image data included in the Fashion-MNIST is a grayscale image comprising 28 pixels in heightand 28 pixels in width, for a total of 784 pixels, and comprised 10 types of fashion products. Each pixelhas a single pixel value indicating the brightness or darkness of the pixel.

Since a digital image is usually created by the intensity of the three primary colors of light, green, red, and blue (GRB), this can be referred to as three luminances being assigned to one pixel. However,in the image data created by grayscale, each element of GRB has the same brightness so an image can be created by giving a single pixel value to one pixel.

Fashion-MNIST is data created for learning neural networks, and the dataset consists of 60,000 images for training (6,000 images for each of 10 types of fashion products) and 10,000 images for testing. Since this study aims to classify the image data, only 60,000 images prepared for training will be used in the verification experiment.

Figure 3 shows the list of 10 product categories included in the Fashion-MNIST data together with sample images.

Convolution of image data and the extraction of a feature vector

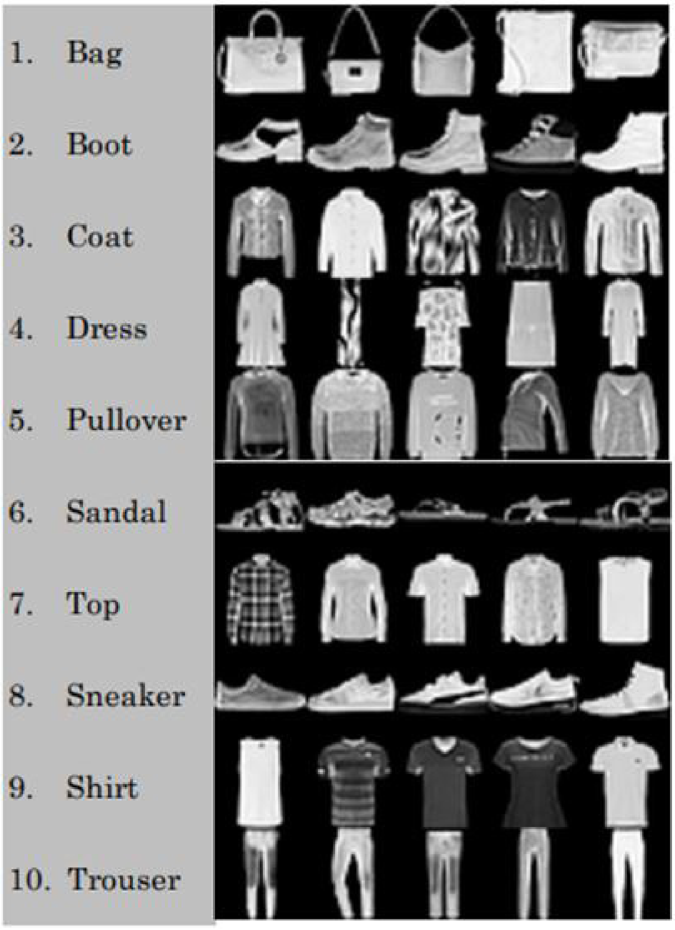


Figure 3: Product categories and sample images included in the Fashion-MNIST

In this section, we describe the results of convolutional processing and extraction of the feature vectorfrom a set of 60,000 image data related to fashion products on the basis of the proposed method in Section 3.1.

Analytical environment

We use the deep learning library Keras developed by Python and executed Tensorflow for convolutional processing.

Overall architecture of convolutional processing

The size of the pooling filter is fixed at 2 × 2, and then, convolutional processing for image data is performed using four types of kernel filters (2 × 2, 3 × 3, 4 × 4, and 5 × 5). We also prepared five typesof filter numbers (16, 32, 64, 128, and 256) and conducted experiments by changing the number of filters. This is to investigate how the classification accuracy changes by changing the size and the number of kernel filters and to determine the optimum size and number of filters.

To improve the analysis accuracy, the convolutional processing is performed twice and then the maxpooling process is performed once. Table 1 shows the results of summarizing the size of the data generated in each processing when the size of the kernel filter is changed. In this study, we will preparefive types of kernel filters to be used for analysis and investigate the effect of the combination of the size and number of kernel filters on the classification results of product categories.

Table 1: Size of data generated when changing the size of a kernel filter

Filter name to use	**Data Status**	**Kernel Filter size**			
		2 x 2	**3 x 3**	**4 x 4**	**5 x 5**
	Input Data (Image Data)	28 x 28	28 x 28	28 x 28	28 x 28
Convolutional Filter1	Convolutional Layer1	27 x 27	26 x 26	25 x 25	24 x 24
Convolutional Filter2	Convolutional Layer2	26 x 26	24 x 24	22 x 22	20 x 20
Pooling Filter1(2x2)	Pooling Layer1	13 x 13	12 x 12	11 x 11	10 x 10
Convolutional Filter3	Convolutional Layer3	12 x 12	10 x 10	8 x 8	6 x 6
Convolutional Filter4	Convolutional Layer4	11 x 11	8 x 8	5 x 5	2 x 2
Pooling Filter 2 (2x2)	Pooling Layer2	5 x 5	4 x 4	2 x 2	1 x 1
	Flatten Layer	5 x 5 x Number of Kernel Filters	4 x 4 x Number of Kernel Filters	2 x 2 x Number of Kernel Filters	1 x 1 x Number of Kernel Filters

Classification of image data via the K-means method

Next, the image data are classified on the basis of the K-means method according to the procedure inSection 3.1 using the vector $\mathbf{A}k$ extracted from the Flatten layer.

Each value of the SSE is recorded while changing the number of clusters to determine the optimum size and number of kernel filters. The value of the log-likelihood chi-square calculated from the actualproduct category and the classification result by cluster analysis is plotted on the graph. Finally, we evaluate the shape of the graph and determine the optimal size and number of kernel filters.

Figures 4_1 to 4_5 show the relationship between the number of clusters and the SSE using a line graph.

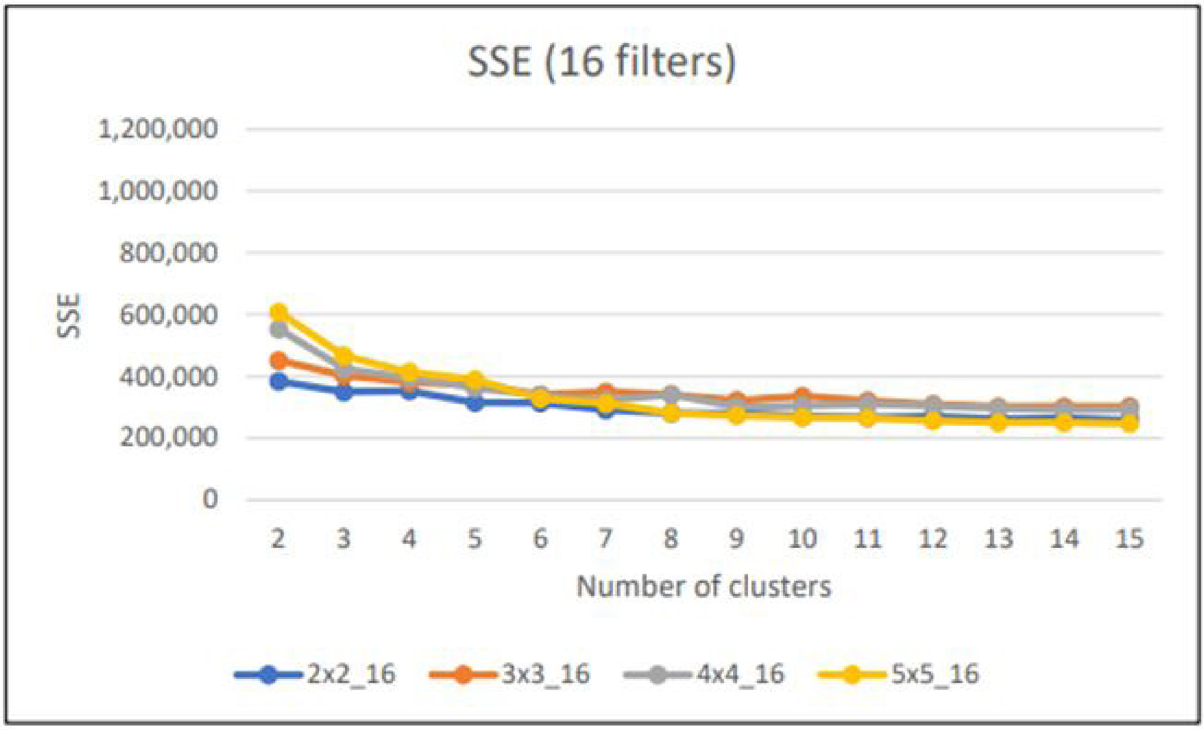


Figure 4.1: Relationship between the number of clusters and SSE (16 kernel filters)

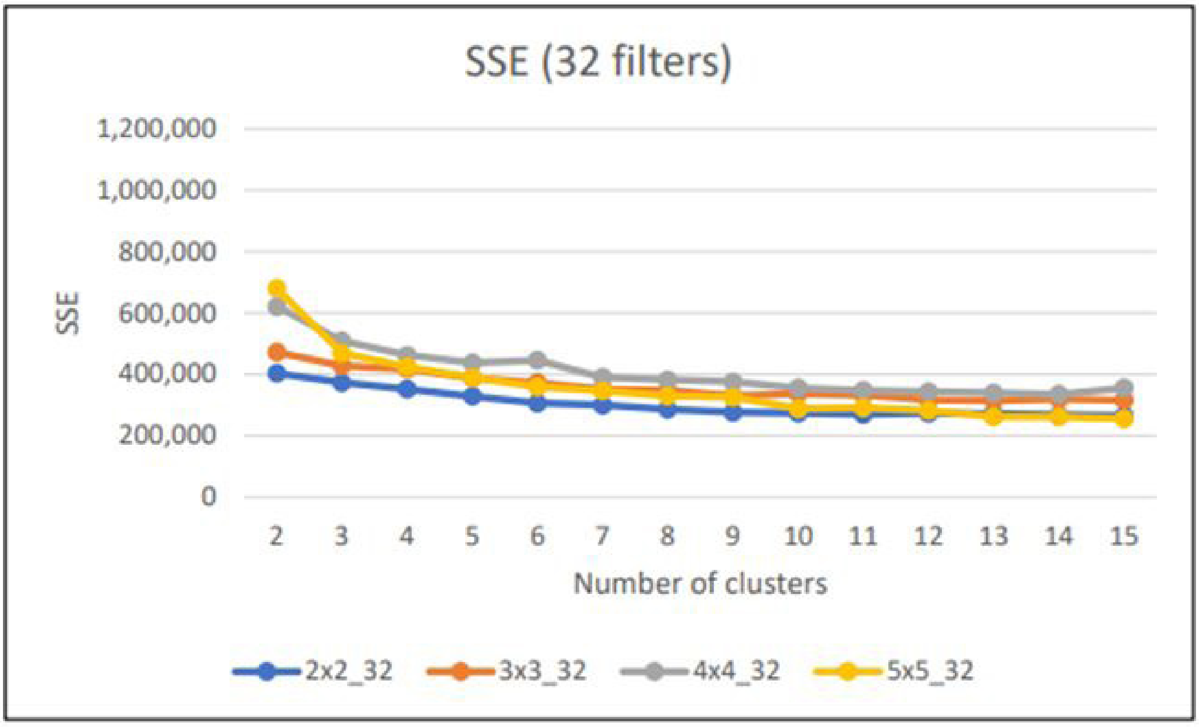


Figure 4.2: Relationship between the number of clusters and SSE (32 kernel filters)

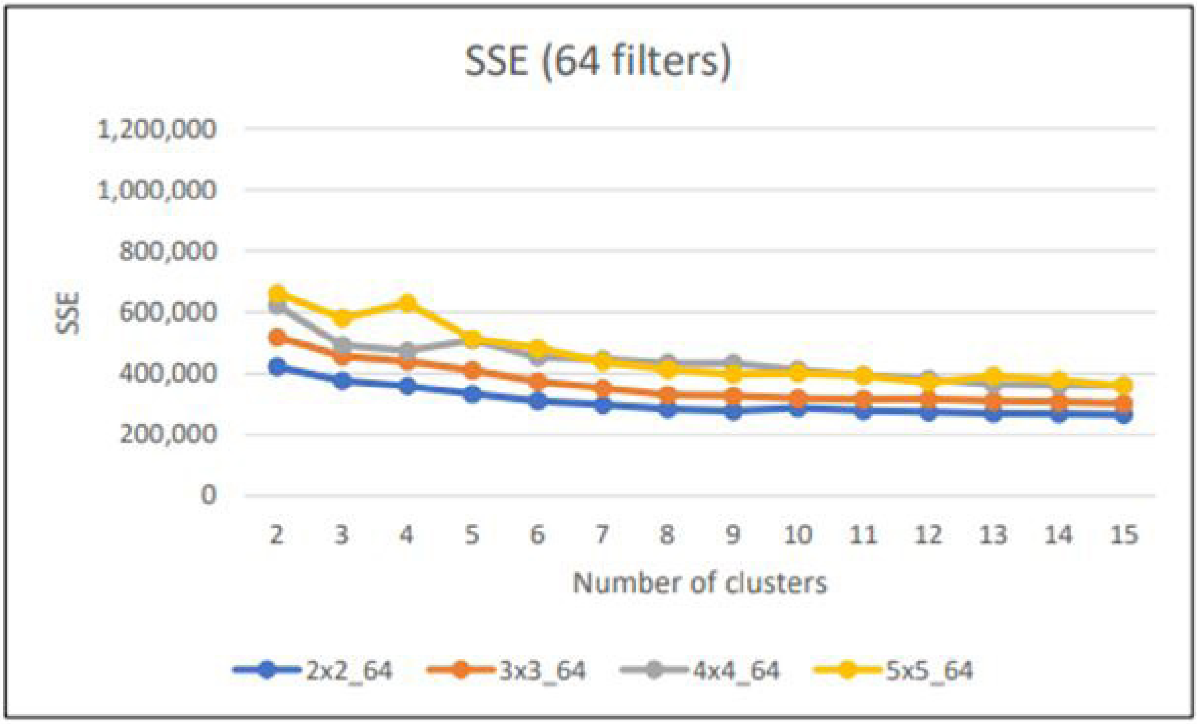


Figure 4.3: Relationship between the number of clusters and SSE (64 kernel filters)

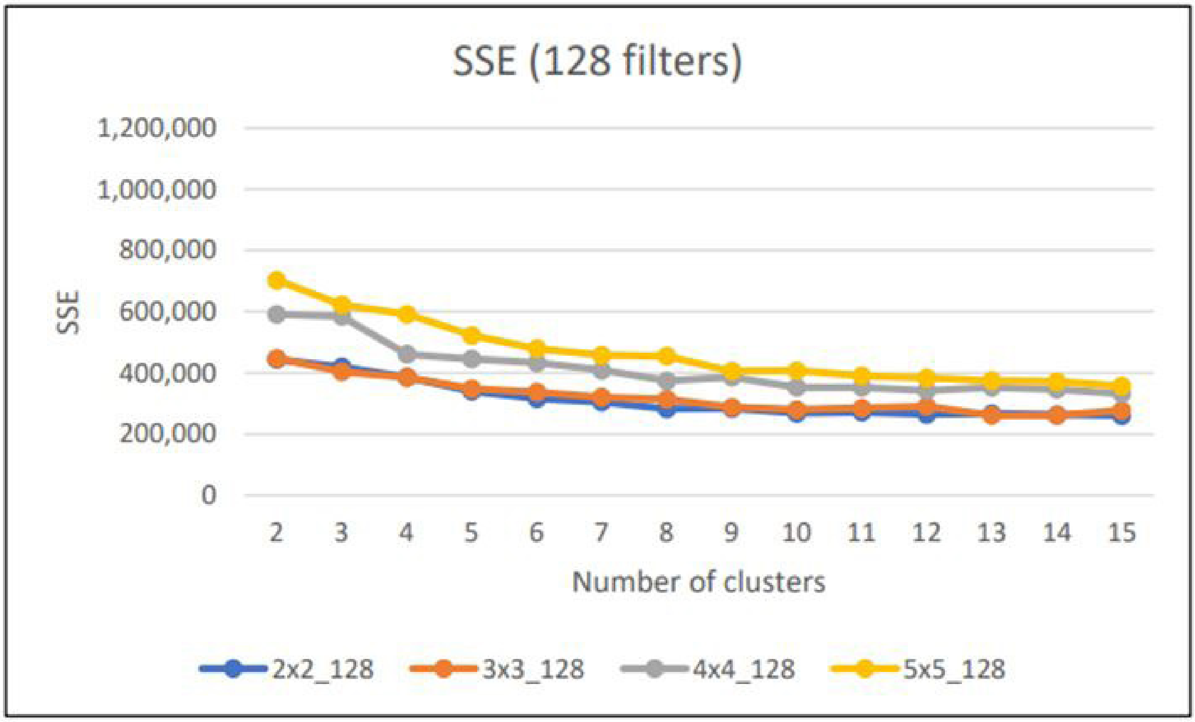


Figure 4.4: Relationship between the number of clusters and SSE (128 kernel filters)

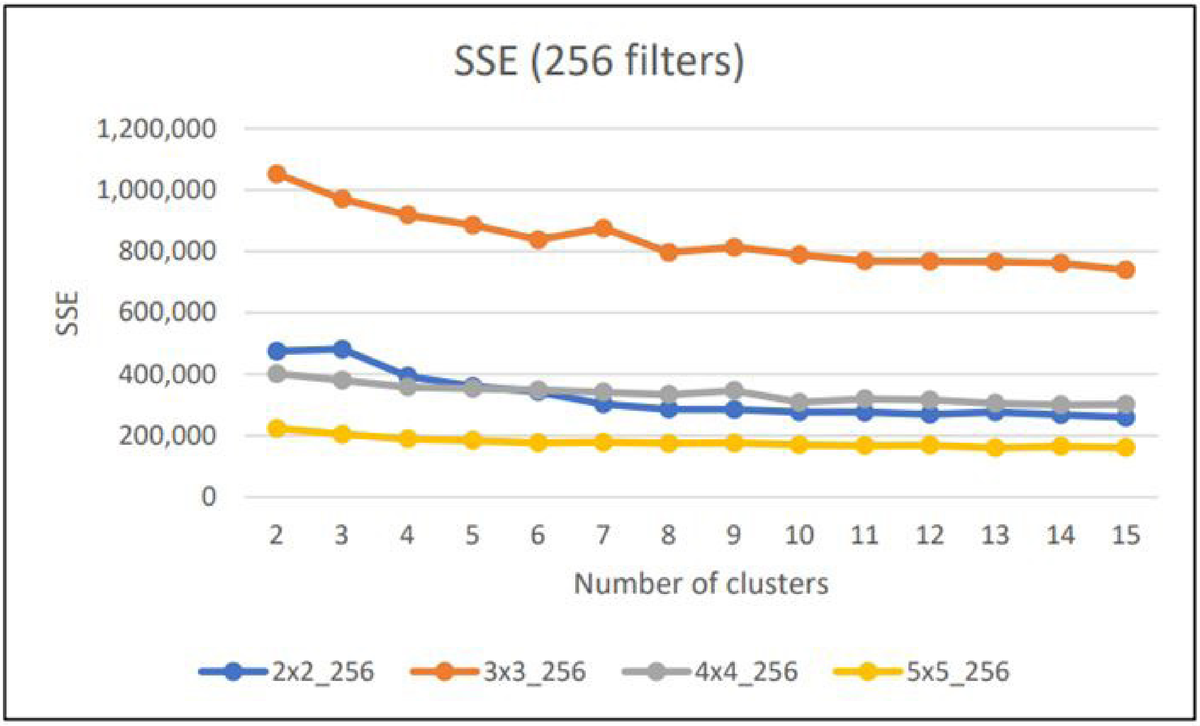


Figure 4.5: Relationship between the number of clusters and SSE (256 kernel filters)

Figure 5.1 to Figure 5.5 show the relationship between the number of clusters and log-likelihood chi-square.

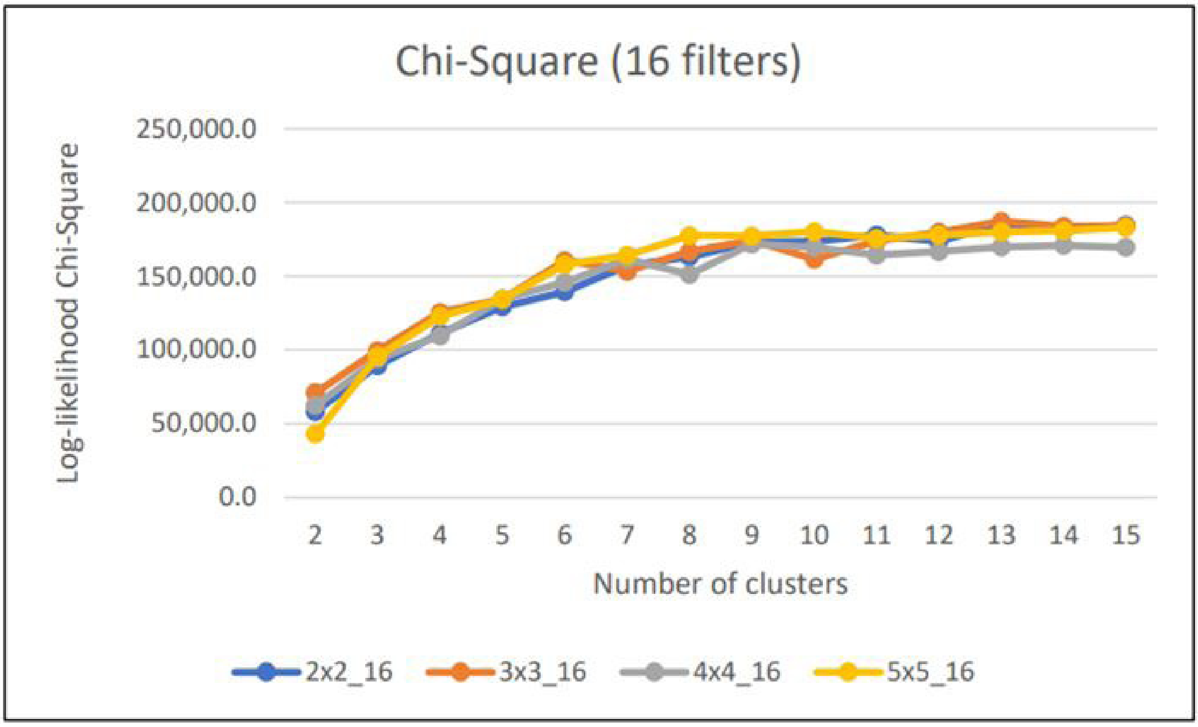


Figure 5.1: Relationship between the number of clusters and log-likelihood chi-square(16 kernel filters)

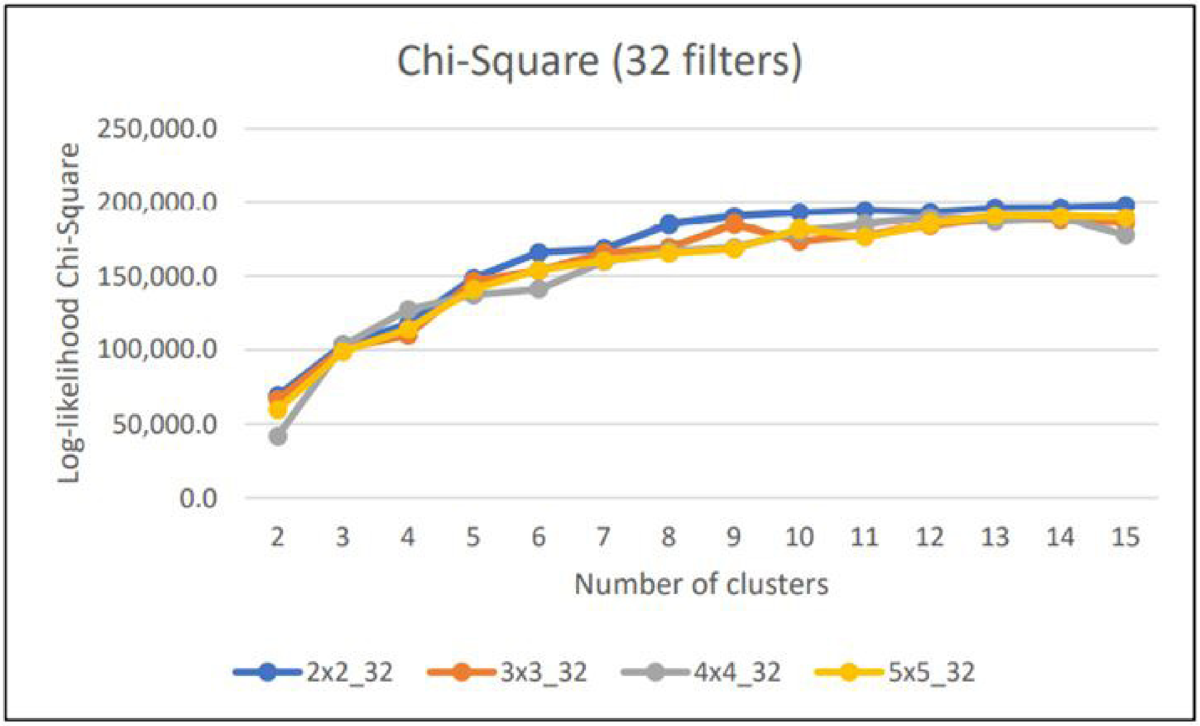


Figure 5.2: Relationship between the number of clusters and log-likelihood chi-square (32 kernel filters)

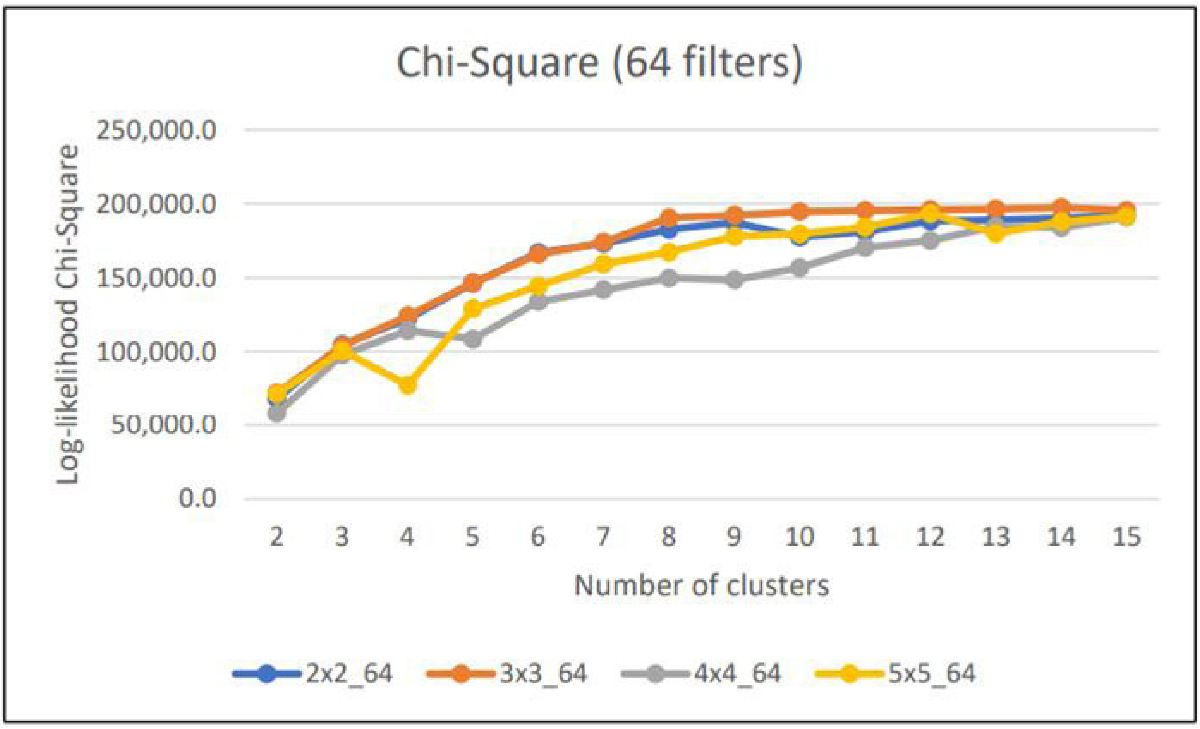


Figure 5.3: Relationship between the number of clusters and log-likelihood chi-square (64 kernel filters)

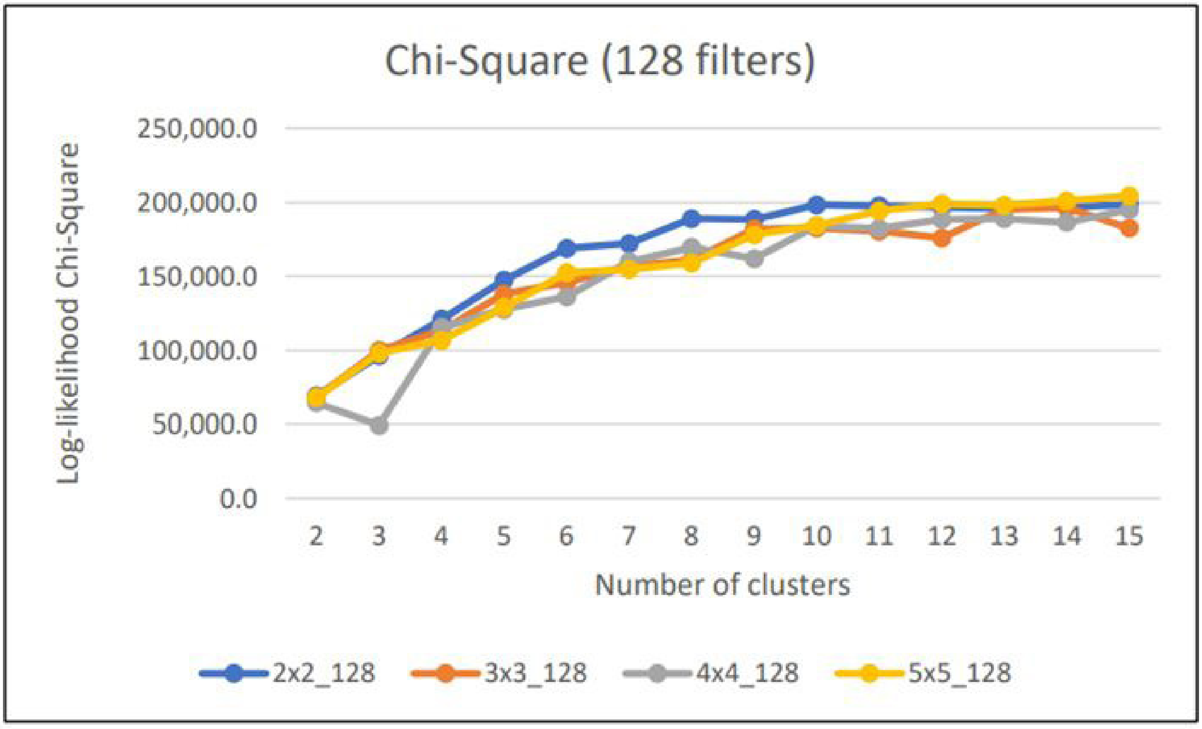


Figure 5.4: Relationship between the number of clusters and log-likelihood chi-square (128 kernel filters)

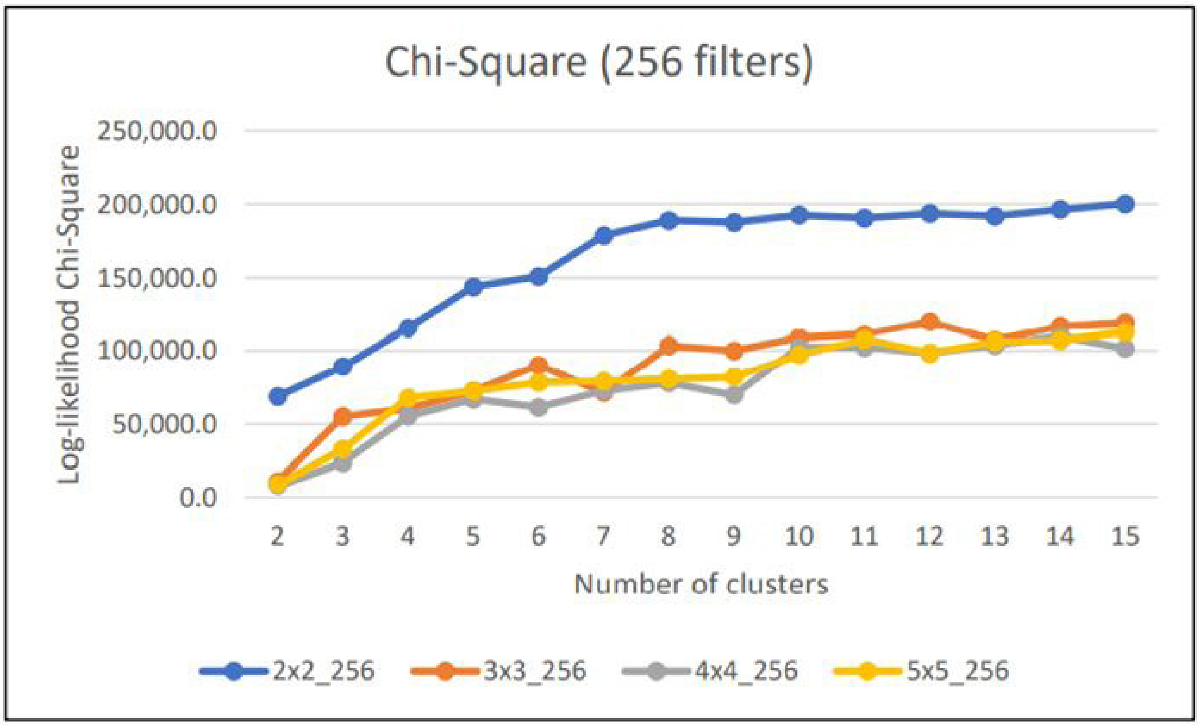


Figure 5.5: Relationship between the number of clusters and log-likelihood chi-square (256 kernel filters)

As a result of the analysis, regarding the number of filters to be used, we can understand that the SSEtends to take a small value regardless of the filter size when the number of filters is small and the SSE value tends to increase as the number of filters increases.

Conversely, regarding the log-likelihood chi-square, it can be seen that the value is higher when thenumber of filters is 32 than when the number of filters is 16, but the value decreases as the number of filters is further increased. Also, looking at the filter size, in the case of a 2 × 2 filter, it can be seen that chi-square takes a large value regardless of the number of filters.

When creating datasets by convolutional processing and performing cluster analysis on the basis of the created data, the time required for data processing and calculation tends to increase as the filter sizeand the number of filters increases.

Considering these points, it can be seen that even if the filter size and the number of filters are increased, the classification accuracy is not necessarily improved.

As a result of verification experiments, when a filter size of 2 × 2 is used, it tends to show stable andhigh classification accuracy. Also, when comparing the case of using 32 filters and the case of using more filters, there is no significant difference in the value of the chi-square, so when classifying the image data of the Fashion-MNIST, it is thought that the optimum filter set would be when the filter sizeis 2 × 2 and the number of filters is 32.

Next, we will examine the optimal number of clusters in this study. Figure 2_2 shows the results of SSE measurements using 32 and 2 × 2 filters. Looking at the graph, we can understand that the numberof clusters continues to decrease monotonically until around 10 and that the SSE has converged after that. Since there are 10 types of fashion products used in the Fashion-MNIST, we will proceed with theanalysis after setting the number of clusters to 10.

Classification Result

Table 2 shows the results of classifying a set of 60,000 image data for trains included in the Fashion-MNIST after setting the size of the kernel filter to 2 × 2, the number of filters to 32, and the number ofclusters to 10.

Classification by k-means clustering

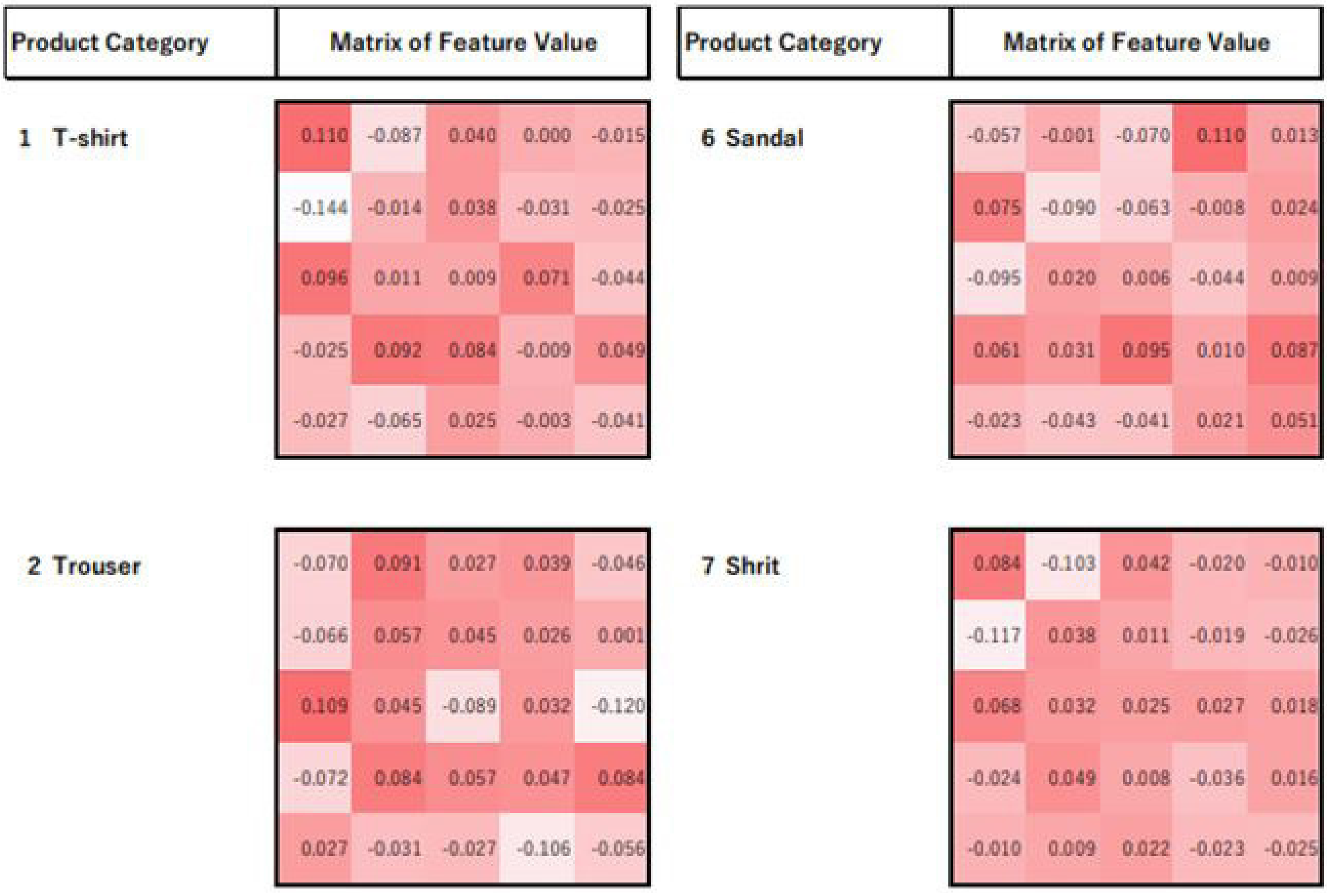
	1	2	3	4	5	6	7	8	9	10	Total
1 T-shirt	90	2	74	8	5,502	2	0	175	30	117	6,000
2 Trouser	1,010	0	13	1	108	4,797	0	69	2	0	6,000
3 Pullover	19	0	56	3	211	0	0	4,178	41	1,492	6,000
4 Dress	3,473	0	3	1	2,223	2	1	222	27	48	6,000
5 Coat	103	0	17	1	680	2	0	5,065	50	82	6,000
6 Sandal	40	2,195	15	3,514	1	0	235	0	0	0	6,000
7 Shrit	53	0	64	0	2,030	0	1	3,384	193	275	6,000
8 Sneaker	0	1	0	4,669	0	0	1,330	0	0	0	6,000
9 Bag	57	0	5,463	172	82	3	42	147	20	14	6,000
10 Ankle boot	45	3	7	43	3	0	5,899	0	0	0	6,000
Total	4,890	2,201	5,712	8,412	10,840	4,806	7,508	13,240	363	2,028	60,000

products representing the cluster	Trouser Dress	Sandal	Bag	Sandal Sneaker	T-shirt Dress Shirt	Trouser	Sneaker Ankle boo	Pullover Coat Shirt	Pullover

In Table 4, the row direction (vertical axis) indicates the product category in which each of the data was originally categorized, and the column direction (horizontal axis) indicates the classification destination of the image data by cluster analysis.

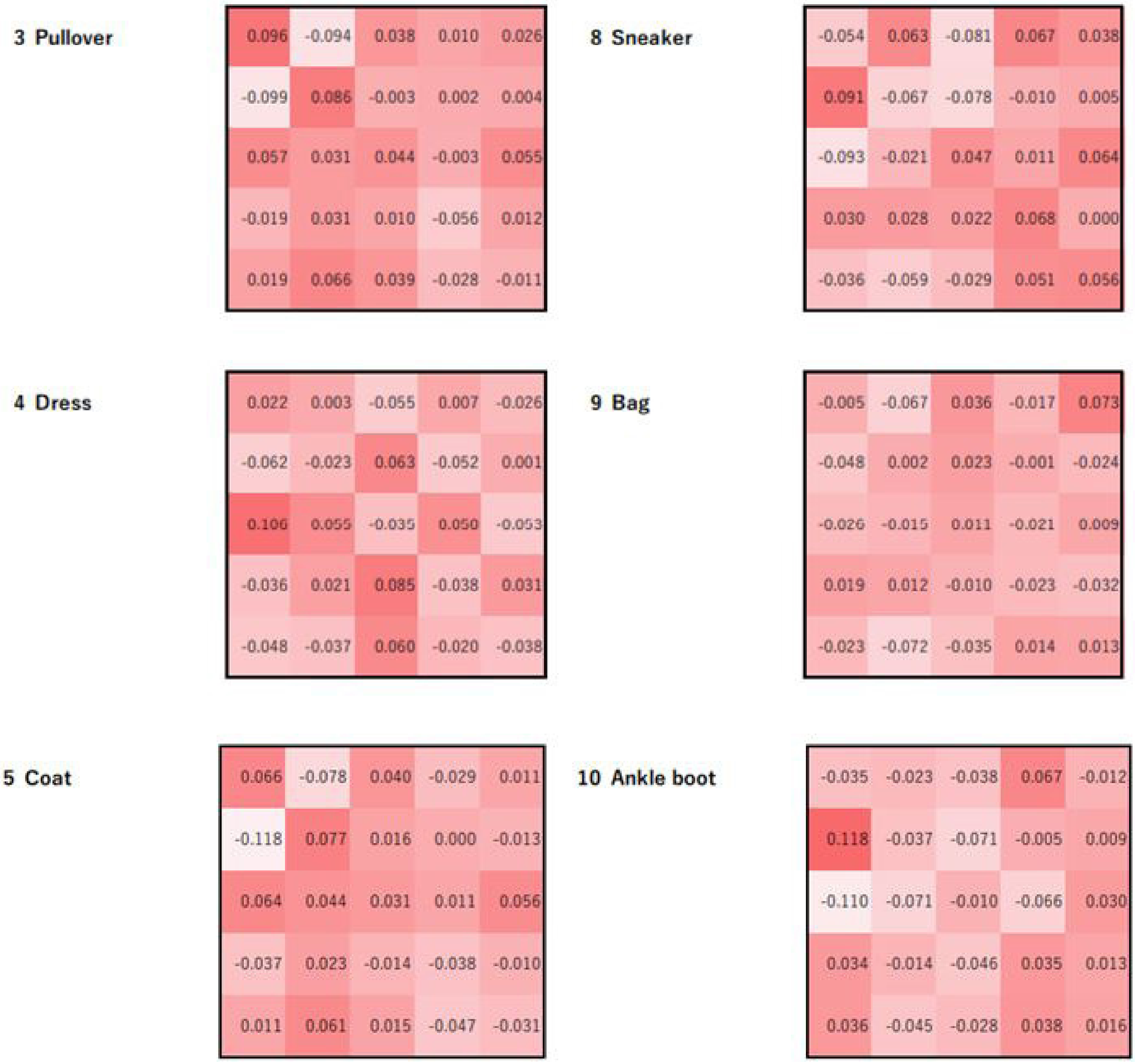
The bottom of Table 2 shows the results of summarizing what kind of products each of the 10 clustersis composed of, on the basis of the classification results.

Figure 6 shows the result of calculating the average value of the feature vector finally extracted using32 of the 2 × 2 kernel filters for each product category. From this result, it can be confirmed that the value of the feature vector extracted by the convolutional processing differs depending on the product category.



Product Category	Matrix of Feature Value				
1 T-shirt	0.110	-0.087	0.040	0.000	-0.015
	-0.144	-0.014	0.038	-0.031	-0.025
	0.096	0.011	0.009	0.071	-0.044
	-0.025	0.092	0.084	-0.009	0.049
	-0.027	-0.065	0.025	-0.003	-0.041
2 Trouser	-0.070	0.091	0.027	0.039	-0.046
	-0.066	0.057	0.045	0.026	0.001
	0.109	0.045	-0.089	0.032	-0.120
	-0.072	0.084	0.057	0.047	0.084
	0.027	-0.031	-0.027	-0.106	-0.056

Product Category	Matrix of Feature Value				
6 Sandal	-0.057	-0.001	-0.070	0.110	0.013
	0.075	-0.090	-0.063	-0.008	0.024
	-0.095	0.020	0.006	-0.044	0.009
	0.061	0.031	0.095	0.010	0.087
	-0.023	-0.043	-0.041	0.021	0.051
7 Shrit	0.084	-0.103	0.042	-0.020	-0.010
	-0.117	0.038	0.011	-0.019	-0.026
	0.068	0.032	0.025	0.027	0.018
	-0.024	0.049	0.008	-0.036	0.016
	-0.010	0.009	0.022	-0.023	-0.025



3 Pullover

0.096	-0.094	0.038	0.010	0.026
-0.099	0.086	-0.003	0.002	0.004
0.057	0.031	0.044	-0.003	0.055
-0.019	0.031	0.010	-0.056	0.012
0.019	0.066	0.039	-0.028	-0.011

4 Dress

0.022	0.003	-0.055	0.007	-0.026
-0.062	-0.023	0.063	-0.052	0.001
0.106	0.055	-0.035	0.050	-0.053
-0.036	0.021	0.085	-0.038	0.031
-0.048	-0.037	0.060	-0.020	-0.038

5 Coat

0.066	-0.078	0.040	-0.029	0.011
-0.118	0.077	0.016	0.000	-0.013
0.064	0.044	0.031	0.011	0.056
-0.037	0.023	-0.014	-0.038	-0.010
0.011	0.061	0.015	-0.047	-0.031

8 Sneaker

-0.054	0.063	-0.081	0.067	0.038
0.091	-0.067	-0.078	-0.010	0.005
-0.093	-0.021	0.047	0.011	0.064
0.030	0.028	0.022	0.068	0.000
-0.036	-0.059	-0.029	0.051	0.056

9 Bag

-0.005	-0.067	0.036	-0.017	0.073
-0.048	0.002	0.023	-0.001	-0.024
-0.026	-0.015	0.011	-0.021	0.009
0.019	0.012	-0.010	-0.023	-0.032
-0.023	-0.072	-0.035	0.014	0.013

10 Ankle boot

-0.035	-0.023	-0.038	0.067	-0.012
0.118	-0.037	-0.071	-0.005	0.009
-0.110	-0.071	-0.010	-0.066	0.030
0.034	-0.014	-0.046	0.035	0.013
0.036	-0.045	-0.028	0.038	0.016

Figure 6: Average value of feature vector finally extracted using 32 of 2 × 2 kernel filters

As a result of the analysis, it can be seen that for each cluster, the products that characterize that cluster are gathered. It turns out that there are cases where an image is classified into the same cluster if the shapes are similar, even if the actual product categories are different, as seen with the pullover, coat, and shirt gathered in cluster number 8.

Classification of image data by comparison method

Here, on the basis of Section 3.2, as a method to be compared with the proposed method in this study,we show the results of clustering by the K-means method on the basis of the original pixel values extracted from the image data.

Figure 7 shows the transition of the likelihood ratio chi-square calculated by changing the number ofclusters from 2 to 15. The solid red line is the result of classification using 32, 2 × 2 kernel filters, and the broken blue line is the result of classification by the method shown in Section 3.2 without convolution of image data.

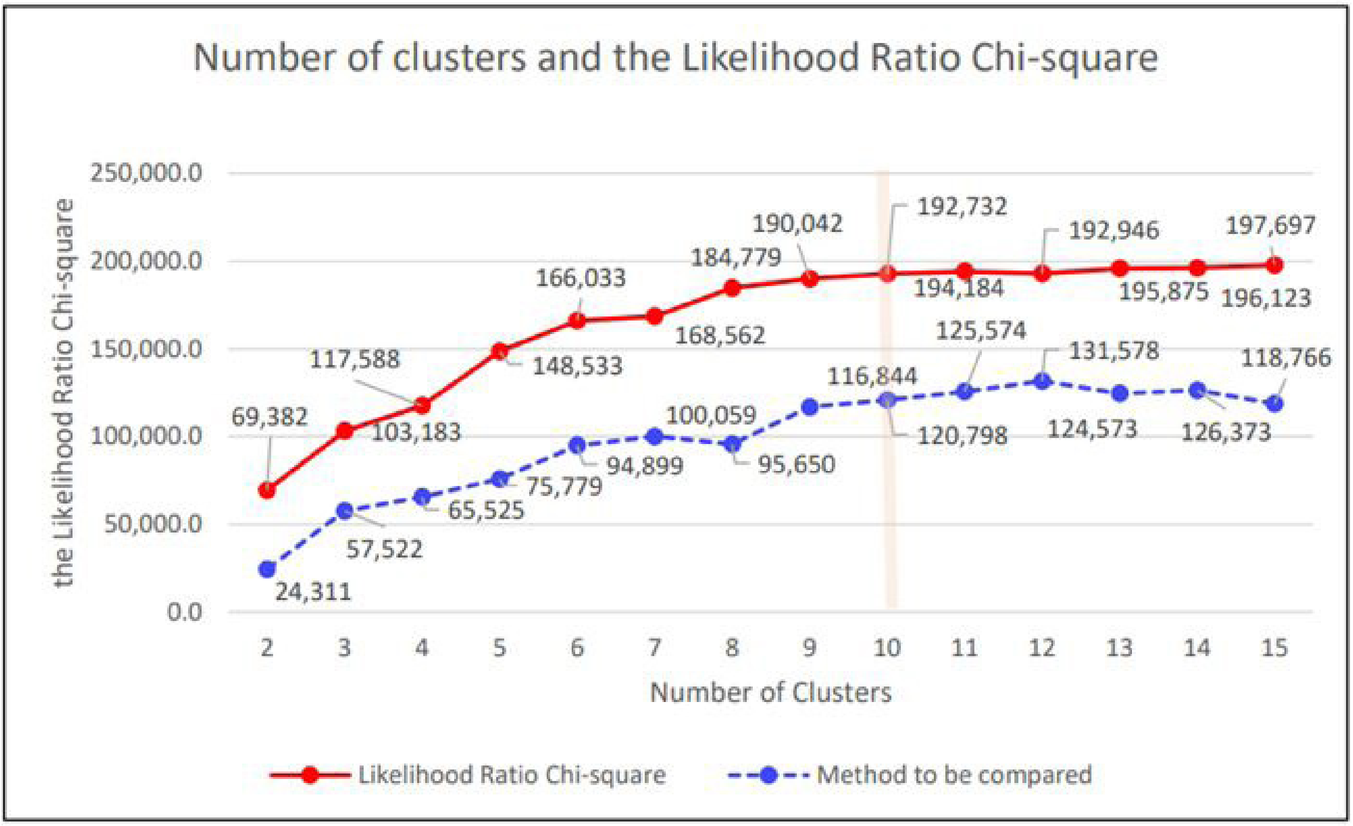


Figure 7: Number of clusters vs likelihood ratio chi-square

From this graph, we can understand that the proposed method in this study shows a higher likelihoodratio chi-square than the comparison method in all the cluster numbers. Especially the likelihood ratio chi-square when the number of clusters is 10 is as follows: Proposed method in this study: 192,732Comparison method: 120,798

It can be seen that the proposed method in this study shows higher classification accuracy than the comparison method.

From the results of this verification experiment, it was shown that image data can be efficiently classified by using the feature vector extracted from convolutional processing. Additionally, it was also shown that the data can be classified with higher accuracy by using the feature vector extracted by the convolutional processing than by using the pixel values.

Conclusion

In this study, we reported the results of studies on the clustering of image data using convolutional processing.

In today's distribution industry, to improve management efficiency, efforts are often being made byclassifying customers or products into groups with similar characteristics and implementing similar marketing strategies inside groups. Under these circumstances, clustering is one of the methods widelyused when classifying customers and products into groups with similar characteristics.

However, most current studies focus on quantifying attributes such as price, size, and color and classifying objects on the basis of the quantified data. With the spread of e-commerce, there are increasing cases where products are introduced by image data instead of displaying details by characterand numeric on the screen.

Considering that most of the product introductions in internet businesses are performed using imagedata, we believe that the clustering of image data can streamline the classification of products for the company that manages e-business. However, almost no research has been conducted to classify data bydirectly performing cluster analysis on image data. In this study, we attempted to propose a method forclassifying products into groups with similar characteristics by convolutional processing for image datarelated to fashion products.

As a result of applying the proposed method to actual data, although there are cases where products with similar shapes are classified into the same cluster even if the original product categories are different, it was confirmed that it is possible to classify similar products into the same cluster to some extent.

The results of the chi-square test show that higher classification accuracy can be obtained in the casewhere the image data are classified by using the feature vector extracted by the convolutional processingthan the case where the image data are classified by using the original pixel value extracted from the data.

Furthermore, in the convolutional processing when extracting the feature vector, even if the numberof filters used in the process is increased, the analysis accuracy does not necessarily improve and the number of filters suitable for the data may exist. Also, regarding the size of the filter to be used, it is possible to classify images with higher accuracy with a filter with a smaller size than with a filter witha larger size and that there is also an appropriate filter size depending on the data.

Finally, we mention the issues of this study and future prospects. In the cluster analysis based on theproposed method, similar products could be classified into the same cluster to some extent. Conversely,there is also a phenomenon in which products belonging to different product categories are grouped together in one cluster. It is believed that one of the causes is that the image data used in this study havea low resolution. And it seems that it is necessary to improve the algorithm so that products are classified into different clusters if the product categories are different even if the images have similar shapes.

Besides product images, e-commerce is also actively promoting using video and audio, so it seems that there is a social need for algorithms that classify videos and audio into groups with similar characteristics.

Acknowledgement

The author would like to thank the anonymous referee who provided useful and detailed comments ona previous/earlier version of the manuscript.

References

1. Willshaw DJ and Von Der Malsburg C (1976) How patterned neural connections can be set up by self-organization, Proceedings of the Royal Society of London. Series B, Biological Sciences 194: 431-445.

2. Anderson TW (1958) An Introduction to Multivariate Statistical Analysis. New York: Wiley

3. Fisher WD (1958) On grouping for maximum homogeneity. Journal of the American Statistical Association 53: 789-798

4. Cox DR (1957) Note on grouping. Journal of the American Statistical Association 52: 543-547.

5. Ward J (1963) Hierarchical grouping to optimize an objective function. Journal of the American Statistical Association 58: 236-244

6. MacQueen JB (1967) Some Methods for classification and Analysis of Multivariate Observations, Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability. University of California Press 281–297

7. Frank RE and Green PE (1968) Numerical Taxonomy in Marketing Analysis: A Review Article, Journal of Marketing Research 1: 83-94

8. George SD and Roger MH (1971), Using cluster analysis to improve marketing experiments, Journal of Marketing Research 8: 340-347.

9. Ian G, Yoshua B and Aaron C (2016) Deep Learning. Cambridge: The MIT Press

10. Fukushima K (1980) Neocognitron: A Self-organizing Neural Network Model for a Mechanism of Pattern Recognition Unaffected by Shift in Position. Biological Cybernetics 36: 193-202

11. Fukushima K (2007) Neocognitron, Scholarpedia 2: 1717.