International Journal of Computer Information Systems and Industrial Management Applications
ISSN 2150-7988 Volume 18 (2026) pp. 1-12

https://doi.org/10.70917/ijcisim-2026-1106
Article

# An Accessory Detection Algorithm Based on Sparse Representation and Low-Rank Matrices

**Xi Jun** [1,*]

[1] Hohhot Minzu College, Hohhot, Inner Mongolia, 010000, China
* Correspondence author: junxi202606@163.com

**Abstract:** To address the issues of low detection accuracy, poor robustness, and limited generalization in accessory detection under complex scenes, we propose an accessory detection algorithm that integrates sparse representation with low-rank matrices. First, discriminative low-rank matrix recovery is employed to correct poor-quality training samples. By learning a low-rank projection matrix, the feature matrix of the test sample is projected onto a corresponding low-rank subspace. These two matrices are then used to construct a redundant dictionary for sparse representation. Classification is performed using the sparse representation approach, and the sparse representation coefficients are obtained by solving the problem with the Adaptive Penalty Linearized Alternating Direction Method (LADMAP). Finally, the reconstruction error for each class is calculated using the sparse representation coefficients, thereby achieving accurate detection of accessories worn by people. The results show that the proposed method achieves an average accuracy of 83.71% with 38.78 million parameters. Compared to the baseline model YOLOv5s, the number of parameters is reduced by 40.88%, while the average accuracy is improved by 16% and the precision is increased by 0.1507, demonstrating excellent performance in accessory detection.

**Keywords:** sparse representation; low-rank matrix; accessory detection algorithm; redundant dictionary; LADMAP

## 1. Introduction

Whether purchasing, investing in, or wearing jewelry on a daily basis, its authenticity directly impacts one's rights and user experience. Traditional jewelry testing typically focuses on three key areas: first, fineness; second, material composition, structure, and craftsmanship; and third, compliance and certification information [1–2]. This applies to individual consumers, retailers, importers, recycling and reprocessing companies, as well as institutions requiring jewelry quality appraisal.

However, with the development of modern society, counterfeiting techniques have become increasingly sophisticated, and the demands on jewelry authentication have grown correspondingly. To address these challenges, authentication algorithms based on sparse representation and low-rank matrix decomposition have garnered significant attention. By leveraging the advantages of sparse representation and low-rank matrix decomposition, these algorithms can effectively address issues in jewelry authentication and improve detection accuracy [3–5]. Sparse representation is a signal processing technique that reduces data dimensions by transforming a signal into a sparse vector [6–7]. In jewelry authentication, jewelry images can be represented as high-dimensional vectors, which are then subjected to sparse representation to reduce the data's dimensionality. Specifically, we can use sparse coding algorithms to perform dimensionality reduction on jewelry images, thereby obtaining their sparse representations [8]. Low-rank matrix decomposition is a linear algebra technique that decomposes a high-dimensional matrix into the product of two low-rank matrices [9–10]. In jewelry detection, low-rank matrix decomposition can be used to process jewelry images, thereby obtaining

their low-rank matrix representations.

This paper proposes a method for representing the features of personal accessories using sparse representation techniques, and employs low-rank decomposition to mitigate the impact of various interfering factors on images and reduce clustering correlations. Furthermore, recognition is achieved by projecting images onto a low-rank subspace and converting them into sparse representations; the use of low-rank decomposition minimizes the influence of noise, thereby enhancing the robustness and reliability of the recognition results. Furthermore, the paper proposes applying the concept of discriminative low-rank matrix recovery to accessory detection. By incorporating a regularization term for ideal coding into the objective function of dictionary learning, category label information is integrated into the dictionary learning process, ensuring that the learned dictionary for sparse representation possesses excellent reconstruction capability and discriminative power. Finally, the method is validated on a person-accessory detection dataset and compared with other detection algorithms to verify its effectiveness.

## 2. Method

### *2.1. Sparse representation*

In the field of image recognition, particularly with regard to jewelry detection and recognition, a set of images of the same test object is organized into a column vector:

$$R_i = \left[ k_{i,1}, k_{i,2}, \cdots, k_{i,n_i} \right] \in K^{m \times n_i} \tag{1}$$

Here, $m$ represents the feature vector dimension of the image, and $k$ represents the set of column vectors for the training set samples. The global vector set of training samples for the object of interest can be expressed as follows:

$$R = \left[ R_1, R_2, \cdots, R_T \right] = \left[ k_{1,1}, k_{1,2}, \cdots, k_{T,n_T} \right] \in K^{m \times M} \tag{2}$$

Here, $M = \sum_{i=1}^{T} n_i$ represents the size of the training set, and $T$ represents the class of the image to be detected. In a purely theoretical context, the aforementioned set of vectors generally follows a low-rank representation. However, in uncontrolled scenarios, due to the influence of various complex interfering factors, the general training set can be represented by $R'$, where $R' = S \vee D$ is the standard low-rank matrix $S \in K^{m \times M}$, combined with the error $D \in R^{m \times M}$ under interference conditions. Once the matrix $D$ follows a Poisson distribution, the calculation of the optimal solution for the low-rank matrix $S$ effectively reduces to solving the following equation:

$$\underset{S,D}{Min} \left\| S \right\| sequence(S) \bullet s, R = S \wedge D \tag{3}$$

Here, $s$ denotes the rank of the low-rank matrix $S$.

Solving the $R$ analysis using traditional principal component analysis (PCA) can theoretically yield an optimal solution; however, in practice, the situation is more complex. Since the $D$ matrix typically does not follow a Poisson distribution, the results obtained from PCA may exhibit significant deviations. For the image objects under analysis, this deviation or error can be represented by a sparse matrix, thereby transforming the problem into a two-objective optimization problem:

$$\underset{S,D}{Min} \left( sequence(S), \left\| D \right\|_0 \right), R = S \wedge D \tag{4}$$

However, the direct consequence of this dual-objective approach is high-dimensional complexity, resulting in a steep decline in the efficiency curve. To address this issue, an optimal weight must be assigned between the two distinct objectives—low-rank quality and sparse error—with the weight parameter denoted by $\lambda$. Consequently, the solution process evolves as follows:

$$\underset{S,D}{Min}\, sequence(S) + \lambda \cdot \left\| D \right\|_0, R = S \wedge D \tag{5}$$

Through multiple iterations, the algorithm gradually converges to the optimal solution.

The optimal mapping matrix for sparse representation is evaluated based on the internal compactness and external sparsity of the vectors. The training dataset $S$ consists of $n$ individual samples, and the column vector representation of each sample in the dataset is denoted as $s_i$, assume

that $s_i$ belongs to Class $x$ of the $S$, and the sample size for that class is $t_x$. The in-group density can be defined as:

$$Min\left\|u_i^k\right\|_1 \ sequence\left\|s_i - S_x u_i^k\right\| < \varphi \tag{6}$$

Specifically, the following two sequences of sets are:

$$S_x = \left[s_{x,1}, s_{x,2}, \cdots, s_{x,n_x}\right] \in K^{m \times n_x} \tag{7}$$

$$u_i^k = \left[u_{i,1}^k, u_{i,2}^k, \cdots, u_{i,n_x}^k\right]^T \in K^{m \times n_x} \tag{8}$$

Here, $u$ represents the in-line sparsity density, and $k$ denotes the in-line method. The in-line sparsity density is expressed as:

$$\sum_{i=1}^{n}\left\|U^T s_i - \sum_{j=1}^{n} P_{ij}^k U^T s_j\right\|^2 = U^T\left[\sum_{i=1}^{n}\left(s_i - SP_i^k\right)\left(s_i - SP_i^k\right)^T\right]U \tag{9}$$

In this context, $P_i^k$ is the $i$ th column vector of $P^k$, and $P^k = \left(P_{ij}^k\right)$ is the inline weight matrix.

The outlier dispersion is represented by a single sample and its exclusive sample set; that is, $s_i$ is reconstructed using the exclusive sample set $S_x$, which excludes $s_i$. The outlier dispersion can be expressed as:

$$Min\left\|u_i^v\right\|_1 \ sequence\left\|s_i - S^x u_i^v\right\|_2 < \varphi \tag{10}$$

Among these, the following two sequences of sets:

$$S^x = \left[S_1, S_2, \cdots, S_i\right] \tag{11}$$

$$u_i^v = \left[u_{i,1}^v, u_{i,2}^v, \cdots, u_{i,n-n_x}^v\right]^T \in K^{n-n_x} \tag{12}$$

Here, $u_i^v$ stands for external link sparsity.

The external link sparsity metric is expressed as:

$$\sum_{i=1}^{n}\left\|U^T s_i - \sum_{j=1}^{n} P_{ij}^v U^T s_j\right\|^2 = U^T\left[\sum_{i=1}^{n}\left(s_i - SP_i^v\right)\left(s_i - SP_i^v\right)^T\right]U \tag{13}$$

In this context, $P_i^v$ is the $i$ th column vector of $P^v$, and $P^v = \left(P_{ij}^v\right)$ is the external weighting matrix.

## *2.2. Low-rank matrix recovery algorithms*

Due to factors such as shadows, reflections, and occlusions, the low-rank structure of the actual acquired jewelry images is disrupted, which affects classification performance. Therefore, effectively recovering the low-rank jewelry subspace from noisy images—that is, removing severe errors in the samples caused by occlusions, lighting, and other factors—is key to improving recognition performance. Low-rank matrix recovery (LRR) is employed to recover a clean dictionary for sparse representation from corrupted training image samples. However, since the most basic LRR algorithms are non-discriminative, they are not suitable for classification tasks. To improve classification performance, a regularization term for ideal coding is added to the dictionary learning objective function, incorporating class label information into the dictionary learning process. This ensures that the dictionary obtained through discriminative low-rank matrix recovery for sparse representation classification is discriminative.

### 2.2.1. Matrix Recovery

Given a data matrix $X = \left[X_1, X_2, \cdots, X_N\right]$ consisting of car logo samples, where $X_i$ corresponds to the $i$ -class sample, $X$ may contain noise. Low-rank matrix recovery decomposes $X$ into a low-rank component $DZ$ and a sparse noise component $E$, i.e., $X = DZ + E$. For the dictionary

$D$, the optimal representation matrix $Z$ of the data matrix $X$ should be a block diagonal matrix:

$$Z^* @ \begin{pmatrix} Z_1^* & 0 & 0 & 0 \\ 0 & Z_2^* & 0 & 0 \\ 0 & 0 & Z_3^* & 0 \\ 0 & 0 & 0 & Z_4^* \end{pmatrix} \tag{14}$$

In particular, dictionary $D=[D_1, D_2, \cdots, D_N]$ contains $N$ subdictionaries, of which $D_i$ corresponds to class $i$. Let $Z_i=[Z_{i,1}, Z_{i,2}, \cdots, Z_{i,N}]$ be the representation coefficient of $X_i$ in dictionary $D$. To obtain a low-rank and sparse data representation, each class $i$ should be well-represented by its sub-dictionary, namely $X_i = D_i Z_{i,i} + E_i$, while $Z_{i,j}$ represents its coefficients in the sub-dictionary $D_j (i \neq j)$, which are nearly zero.

Let $Q$ be the ideal representation coefficient, $Q=[q_1, q_2, \cdots, q_T] \in R^{K \times T}$, and let $q_i$ be the representation coefficient of sample $x_i$, which is of the form $[0 \cdots 1,1,1, \cdots]' \in R^k$, where $K$ is the dictionary size and $T$ is the total number of samples. Assuming that $x_i$ belongs to class $L$, all of its coefficients in sub-dictionary $D_L$ are 1, and all of its coefficients in other sub-dictionaries are 0.

For example, suppose the dictionary $D=[d_1, d_2, \cdots, d_6]$ contains 6 atoms, and the data matrix $X=[X_1, X_2, X_3]$ contains 3 sample classes. If the first class $X_1$ contains 3 samples $x_1, x_2, x_3$, and the second class $X_2$ contains 2 samples $x_4, x_5$, the third category of $X_3$ contains 4 samples of $x_6, x_7, x_3, x_9$, so $D$ has 3 subdictionaries, each with two atoms. Therefore, $Q$ is represented as:

$$Q = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{pmatrix} \tag{15}$$

Based on the above definition, construct a block-diagonal matrix $Q$ for the training samples using label information. Add a regularization term $\|Z-Q\|_F^2$ to incorporate structural information into the dictionary training. The objective function for dictionary learning can then be written as:

$$\begin{aligned} &\min_{Z,E,D} \|Z\|_* + \lambda \|E\|_1 + \beta \|Z\|_1 + \alpha \|Z-Q\|_F^2 \\ &s.t. \quad X = DZ + E \end{aligned} \tag{16}$$

In particular, $\lambda$ and $\beta$ control the sparsity of the noise matrix $E$ and the coefficient matrix $Z$, respectively; $\alpha$ is a regularization term for the adjustment factor; $\|\bullet\|_*$ and $\|\bullet\|_1$ represent the nuclear norm and $l_1$ norm of the matrix, respectively.

This paper employs the adaptive penalty linearized alternating direction method (LADMAP) to solve the optimization problem of the objective function described above. To solve this optimization problem, an auxiliary variable $W$ is introduced to make the objective function separable, so that the objective function can be written as:

$$\begin{aligned} &\min_{Z,E,D} \|Z\|_* + \lambda \|E\|_1 + \beta \|W\|_1 + \alpha \|W-Q\|_F^2 \\ &s.t. \quad X = DZ + E, W = Z \end{aligned} \tag{17}$$

To solve for $Z$ and $E$ using LADMAP, the augmented Lagrangian function can be written as:

$$L(Z,W,E,D,Y_1,Y_2,\mu)=\|Z\|_*+\lambda\|E\|_1+\beta\|W\|_1+\alpha\|W-Q\|_F^2$$
$$+h(Z,W,E,D,Y_1,Y_2,\mu)-\frac{1}{2\mu}\left(\|Y_1\|_F^2+\|Y_2\|_F^2\right) \tag{18}$$

Specifically:

$$h(Z,W,E,D,Y_1,Y_2,\mu)=\frac{\mu}{2}\left(\left\|X-DZ-E+\frac{Y_1}{\mu}\right\|_F^2+\left\|Z-W+\frac{Y_2}{\mu}\right\|_F^2\right) \tag{19}$$

Optimize the function by updating one of the $Z,W,E$ variables while fixing the other two each time:

$$Z^{j+1}=\arg\min_Z\frac{1}{\eta\mu}\|Z\|_+$$
$$+\frac{1}{2}\left\|Z-Z^j+\left[-D^T(X-DZ^j-E^j+\frac{Y_1^j}{\mu})+\left(Z-W_j+\frac{Y_2^j}{\mu}\right)\right]/\eta\right\|_F^2 \tag{20}$$

$$W^{j+1}=\arg\min_W\frac{\beta}{2\alpha+\mu}\|W\|_1$$
$$+\frac{1}{2}\left\|W-\left(\frac{2\alpha}{2\alpha+\mu}Q+\frac{1}{2\alpha+\mu}Y_2^j+\frac{\mu}{2\alpha+\mu}Z^{j+1}\right)\right\|_F^2 \tag{21}$$

$$E^{j+1}=\arg\min_E\frac{\lambda}{\mu}\|E\|_1+\frac{1}{2}\left\|E-\left(\frac{1}{\mu}Y_1^j+X-DZ^{j+1}\right)\right\|_F^2 \tag{22}$$

Specifically, $\eta=\|D\|_2^2$.

low-rank matrix recovery optimization algorithm is as follows:

Input: Data $X$, dictionary $D$, and parameters $\lambda,\beta$ and $\alpha$;

Output: $Z,E,W$;

Initialization: $Z^0=W^0=E^0=Y_1^0=Y_2^0=0,\beta=1.1,\varepsilon=10^{-7},\mu_{\max}=10^{30}$ executes the following loop until the termination condition (not converged, $Z\bullet\max IterZ$) is met;

Using equation (19), fix $W,E$ and update $Z$;

Using equation (20), fix $Z,E$ and update $W$;

Using equation (21), fix $Z,W$ and update $E$;

Update multipliers $Y_1^{j+1}=Y_1^j+\mu\left(X-DZ^j-E^j\right)$ and $Y_2^{j+1}=Y_2^j+\mu\left(Z^j-W^j\right)$;

Update $\mu$: $\mu=\min\left(\mu_{\max},\rho\mu\right)$;

Check convergence condition: $\left\|X-DZ^j-E^j\right\|_\infty<\varepsilon,\left\|Z^j-W^j\right\|<\varepsilon$;

End.

After recovery via low-rank matrix discrimination, the training samples can be divided into two parts: the corrected samples and the sparse error. The corrected samples are used as a dictionary for sparse representation, offering greater discriminative power and reconstruction capability than the original training samples.

## 2.2.2. Matrix Mapping

The discriminative low-rank matrix recovery method described above yields a clean sparse representation dictionary for jewelry with good reconstruction quality and discriminative power; however, the jewelry samples collected for identification often contain noise as well. To address this, this paper proposes learning a low-rank projection matrix to correct test samples with disrupted low-rank structures by projecting them onto the corresponding underlying subspace. In this paper, we assume that $P$ is a low-rank projection matrix that projects data $X$ onto the corresponding underlying subspace, yielding a low-rank subspace reconstruction result $PX$. The mapping matrix $P$ can be obtained by solving the following optimization problem:

$$\min_{P} \|P\|_* \quad s.t. Y = PX \tag{23}$$

Assuming that $P^*$ is the optimal solution to the above problem, calculating $P^*X$ and $X - P^*X$ separately will yield the main component and the error term of the sample $X$.

The optimal solution to the above objective function is $P^* = YX^+$, where $X^+$ is the pseudoinverse of $X$. The pseudoinverse is defined as $X^+ = V\sum^{-1}U^T$, where $X$ is the singular value decomposition (SVD) of $V\sum U^T$.

## *2.3. Construction of Sparse Representation Classifiers*

Sparse representation models seek the sparsest representation of samples in the original input space. This technique maps features that are non-linearly separable in the input space to a high-dimensional feature space via a kernel function, thereby enabling linear separation of similar features within that space.

In this section, we will use the redundant dictionary $H = [A \quad E]$ to obtain the sparse representation coefficients $\alpha \in R^{2n}$ by solving the kernel sparse representation problem, thereby constructing a classifier to classify test samples.

Let the test sample be $yR^d$. The sparse representation problem is to find the sparsest solution $\alpha$ for vector $y$ under dictionary $H = [h_1, h_2, \cdots, h_{2n}]$, which can be formulated as:

$$\begin{cases} \min\limits_{\alpha} & \|\alpha\|_0 \\ s.t. & \|y - H\alpha\|_2^2 \bullet \varepsilon \end{cases} \tag{24}$$

Common methods for solving the above problems include the OMP algorithm, the ROMP algorithm, the BP algorithm, and the SLO algorithm.

Let the nonlinear mapping $\Phi : R^d \to F \subset R^N (d \ll N), F$ have a kernel space that is a Hilbert space. Then the kernel function corresponding to the nonlinear mapping $\Phi$ is expressed as:

$$k(x, x') = \langle \Phi(x), \Phi(x') \rangle = \Phi(x)^T \Phi(x') \tag{25}$$

Among them, $x, x' \in R^d$.

The sparse representation problem can be formulated as follows:

$$\begin{cases} \min\limits_{\alpha} & \|\alpha\|_0 \\ s.t. & \|\Phi(y) - \Phi(H)\alpha\|_2^2 \bullet \varepsilon \end{cases} \tag{26}$$

Among them, $\Phi(H) = [\Phi(H)_1, \Phi(H)_2, \cdots, \Phi(H)_{2n}]$.

In this paper, we will use a transformation matrix to transform the kernel sparse representation problem into a general sparse representation problem, and then solve it using the OMP algorithm. Let the transformation matrix be:

$$P = \Phi(H)B \tag{27}$$

Let $B \in R^{2n \times 2d}$ be a random matrix; then Equation (26) becomes:

$$\begin{cases} \min\limits_{\alpha} & \|\alpha\|_0 \\ s.t. & \|P^T\Phi(y) - P^T\Phi(H)\alpha\|_2^2 \bullet \varepsilon \end{cases} \tag{28}$$

Substituting equation (27) into equation (28) yields:

$$\begin{cases} \min\limits_{\alpha} & \|\alpha\|_0 \\ s.t. & \|B^T\Phi^T(H)\Phi(y) - B^T\Phi^T(H)\Phi(H)\alpha\|_2^2 \bullet \varepsilon \end{cases} \tag{29}$$

Let:

$$
\begin{aligned}
K &= \Phi^T(H)\Phi(H) \\
K_1 &= \Phi^T(H)\Phi(y) \\
&= \left[k\left(h_1, y\right), k\left(h_2, y\right), \cdots, k\left(h_{2n}, y\right)\right]^T
\end{aligned} \tag{30}
$$

Equation (29) can be expressed as:

$$
\begin{cases}
\min\limits_{\alpha} & \|\alpha\|_0 \\
s.t. & \left\|B^T K_1 - B^T K\alpha\right\|_2^2 \bullet \varepsilon
\end{cases} \tag{31}
$$

The minimization problem in Equation (31) is a standard sparse representation problem. In this paper, we use the OMP algorithm to solve it and obtain the sparse representation coefficients $\alpha$.

By applying the OMP algorithm to Equation (31), we obtain the sparse representation coefficients $\hat{\alpha}$, which consist of two parts: the coefficients $\hat{\alpha}_A \in R^n$ corresponding to the basis dictionary $A$ and the coefficients $\hat{\alpha}_E \in R^n$ corresponding to the error dictionary $E$, i.e., $\hat{\alpha} = \left[\hat{\alpha}_A \hat{\alpha}_E\right]^T$.

Therefore, vectors $\hat{\alpha}_A$ and $\hat{\alpha}_E$ can be obtained from vector $\hat{\alpha}$, yielding the reconstruction error between the test sample $y$ and the training samples associated with each of the $i$ classes as:

$$
\begin{aligned}
r_i(y) &= \left\|\Phi(y) - \Phi(A)\delta_i\left(\hat{\alpha}_A\right) - \Phi(E)\hat{\alpha}_E\right\|_2^2 \\
&= \Phi^T(y)\Phi(y) - \Phi^T(y)\Phi(A)\delta_i\left(\hat{\alpha}_A\right) \\
&\quad -\Phi^T(y)\Phi(E)\hat{\alpha}_E - \delta_i^T\left(\hat{\alpha}_A\right)\Phi^T(A)\Phi(y) \\
&\quad +\delta_i^T\left(\hat{\alpha}_A\right)\Phi^T(A)\Phi(A)\delta_i\left(\hat{\alpha}_A\right) \\
&\quad +\delta_i^T\left(\hat{\alpha}_A\right)\Phi^T(A)\Phi(E)\hat{\alpha}_E \\
&\quad -\hat{\alpha}_E^T\Phi^T(E)\Phi(y) \\
&\quad +\hat{\alpha}_E^T\Phi^T(E)\Phi(A)\delta_i\left(\hat{\alpha}_A\right) \\
&\quad +\hat{\alpha}_E^T\Phi^T(E)\Phi(E)\hat{\alpha}_E
\end{aligned} \tag{32}
$$

Using the reconstruction error $r_i(y)$, select the class with the smallest reconstruction error to determine the class to which the test sample $y$ belongs:

$$
indentity(y) = \arg\min_i r_i(y) \tag{33}
$$

Here, $indentity(y)$ represents the class label of test sample $y$.

## *2.4. Jewelry Inspection Algorithm*

Based on the above definitions, the basic workflow of the joint sparse representation and low-rank matrix recovery-based jewelry detection algorithm is described as follows:

Input: Training dataset $D \in R^{d\times n}$, test dataset $y \in R^{d\times n}$;

Output: Class labels $indentity(y) = \arg\min_i r_i(y)$;

Step 1: Use RPCA to learn the basis dictionary $A$ and the error dictionary $E$, which together form the redundancy dictionary $H$;

Step 2: Use the IALM algorithm to solve the optimization problem and obtain the low-rank matrix $A_i$ and the error matrix $E_i$ for each class;

Step 3: Use $A_i$ and $E_i$ to obtain the basis dictionary $A = \left[A_1, A_2, \cdots, A_k\right]$ and the error dictionary $E = \left[E_1, E_2, \cdots, E_k\right]$;

Step 4: Construct the redundant dictionary $H = \left[A \quad E\right]$ and normalize each atom;

Step 5: Use the redundant dictionary $H$ to construct a kernel sparse representation classifier;

Step 6: Solve the kernel sparse representation problem using the OMP algorithm to obtain the coefficients $\hat{\alpha} = \left[\hat{\alpha}_A \quad \hat{\alpha}_E\right]^T$;

Step 7: Calculate the reconstruction error $r_i(y)$ associated with each class.

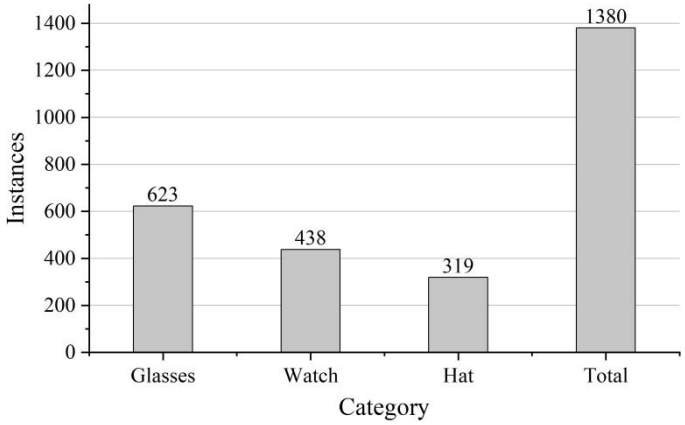
Kernel Selection: In this paper, the Gaussian kernel (also known as the radial basis function) is selected, defined as:

$$k(x, x') = \exp\left(-\gamma \left\| x - x \right\|\right) \tag{34}$$

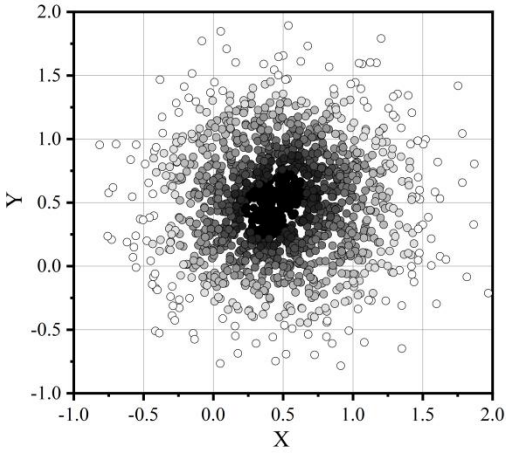
## 3. Experiments and Analysis of Results

### *3.1. Data Preprocessing*

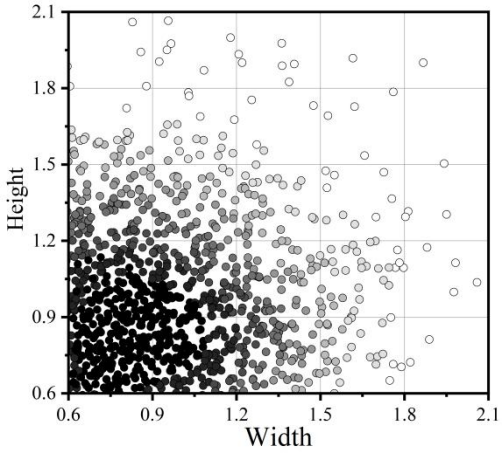
In this study, a custom dataset was used for training and testing. The custom dataset includes three categories: watches, glasses, and hats. Specifically, it contains 623 images of glasses, 438 images of watches, and 319 images of hats. The number of image categories in the dataset and the distribution of annotation bounding boxes are shown in Figure 1. The distribution of image counts across the three categories is shown in Figure 1(a). The category distribution was designed with careful consideration of size and shape variations across different viewpoints, as well as various lighting conditions and situations where objects are obscured, thereby further improving the model's robustness and accuracy. The dataset comprises 1,380 images in JPG format, each with a resolution of 1920×1080, divided into training and testing sets in an 8:2 ratio. The data follows the VOC dataset format, and ground-truth bounding boxes were annotated for each image using the LabelImg annotation tool.




(a) Distribution of image quantity




(b) Distribution map of the centers of image anchor boxes




(c) Distribution map of image anchor box sizes

***Figure 1.*** The number of image categories in the dataset and the distribution of image annotation anchor boxes

To better understand the distribution of images and bounding boxes in the dataset, an analysis was conducted. The distribution of images across the three main accessory categories is relatively even, with no outliers. For smaller, harder-to-recognize object categories such as eyeglasses and watches, images from these two categories account for 76.88% of the total. The distribution of anchor box center positions is relatively uniform, with most concentrated in the lower-middle region of the images, as shown in Figure 1(b). Regarding anchor box sizes, small and medium-sized boxes predominate overall, as shown in Figure 1(c), indicating that the experimental dataset is well-suited for simulating real-world person accessory detection tasks.

To enhance data diversity and prevent model overfitting, data augmentation was performed using the Mosaic and Cutout methods. Additionally, during training, the original dataset was preprocessed through operations such as symmetric flipping, adjusting contrast and brightness, adding noise, and resizing to enrich the dataset. The specific operations are shown in Table 1, including symmetric flipping, adjusting contrast and brightness, adding noise, and resizing.

***Table 1.*** Dataset expansion operations

| Operation | Implementation method |
|---|---|
| Symmetrical flipping | Flip 180 degrees along the Y-axis |
| Change the contrast and brightness | Add a luminance contrast conversion with a Gamma parameter of 2 |
| Add noise | Gaussian noise obeys (0,1) |
| Size scaling | Black border Padding with Padding =24 |

## *3.2. Test Environment*

The experiment utilized the Auto DL cloud computing platform for training and testing; the system configuration is shown in Table 2.

***Table 2.*** Experimental environment

| Name | Related configuration |
|---|---|
| CPU | 16 cores Intel(R) Xeon(R) Platinum 8350c CPU @ 2.60GHz |
| Memory | 32 GB |
| GPU | RTX 3090Ti, video memory :12GB |
| GPU accelerator | CUDA 11.2, CUDANN 8.2.5 |
| Operating system | Ubuntu 20.04 |
| Software environment | PyTorch 1.10.0, Python 3.8 |

## *3.3. Analysis of Results*

### 3.3.1. Ablation Experiment

The jewelry detection algorithm used in this experiment was improved using sparse representation (SRC), low-rank matrix recovery (LRR), adaptive penalty linearized alternating direction method (LADMAP), and the OMP algorithm, while the loss function was also optimized. Ablation experiments were designed to evaluate the extent to which modifications to different modules and combinations of modules optimize algorithm performance. The baseline model for the experiments was YOLOv5s. To ensure experimental accuracy, each set of experimental models used the same training parameters and dataset. Each set of experiments was conducted three times, and the results were averaged for comparison. The data for the ablation experiments are shown in Table 3.

***Table 3.*** Comparison results of ablation experiments

| Experiment Serial Number | SRC | LRR | LADMAP | OMP | Acc | mAP/% | Parameter quantity |
|---|---|---|---|---|---|---|---|
| 1 | – | – | – | – | 0.8254 | 67.71 | 655980 |
| 2 | √ | – | – | – | 0.8211 | 63.13 | 384105 |
| 3 | – | √ | – | – | 0.8704 | 70.48 | 737407 |
| 4 | – | – | √ | | 0.9150 | 76.30 | 1069380 |
| 5 | – | – | – | √ | 0.8991 | 73.16 | 644549 |
| 6 | √ | √ | – | – | 0.9450 | 69.73 | 351110 |
| 7 | √ | – | √ | – | 0.9327 | 78.75 | 622691 |
| 8 | √ | – | – | √ | 0.9142 | 67.08 | 324809 |
| 9 | – | √ | √ | – | 0.9206 | 79.92 | 1017826 |
| 10 | √ | √ | √ | – | 0.9537 | 81.02 | 614239 |
| 11 | – | √ | √ | √ | 0.9522 | 80.45 | 1045369 |
| 12 | √ | √ | √ | √ | 0.9761 | 83.71 | 387843 |

Experiment 2 shows that after incorporating a sparse representation module into the jewelry

detection algorithm, the total number of model parameters decreased from 655,980 to 384,105—a 41.45% reduction—but both the model's mean accuracy and precision declined. Experiments 3–5 demonstrate that when used individually, all three optimization methods can improve the model's mAP. Specifically, adding the low-rank matrix recovery module resulted in a slight increase in the number of parameters, while mAP improved by 2.77% and accuracy by 0.045%; the LADMAP and OMP algorithms improved mAP by 8.59% and 5.45%, respectively.

Based on the results of experiments combining different modules, it was found that using various combinations yields a positive optimization effect on the model's overall performance. In Experiment 9, using the native backbone network, mAP improved by 12.21% and accuracy by 0.0952 after optimizing LRR and LADMAP, although the number of parameters increased significantly. Most of this increase in parameters came from the LADMAP module. In Experiment 11, low-rank matrix recovery was added to the approach used in Experiment 9, resulting in a 0.53% increase in mAP compared to Experiment 9. This demonstrates that combining the three optimization strategies can enhance model performance.

To reduce the number of model parameters while improving mAP and precision, experiments were conducted after applying sparse representations. The experimental results show that different module combinations yield varying optimization effects. Experiments 10 and 12 employed multiple optimization strategies simultaneously; while the results indicate that this approach may diminish the improvement achieved by any single strategy, it achieved the optimal overall performance. In particular, Experiment 12 achieved a 16% increase in mAP and a 0.1507 increase in precision compared to the baseline model, while reducing the total number of parameters by 40.88%, resulting in the most ideal outcome in this experiment.

### 3.3.2. Comparative experiment

This paper selects four traditional image recognition algorithms—Support Vector Machines (SVM), Maximum Likelihood Classification (MLC), K-Nearest Neighbors (KNN), and Low-Rank Matrix Recovery combined with the SRC classifier (LRR-SRC)—as comparison algorithms for experimental validation, in order to demonstrate the effectiveness of the proposed algorithm. Since each image in the jewelry image dataset has a size of 1920×1080, converting them into vectors results in a dimension as high as 2,073,600, resulting in a "curse of dimensionality." In this case, Principal Component Analysis (PCA) is employed for dimensionality reduction, with reduction levels set at 20, 50, 100, 200, and 400 dimensions. Table 4 compares the detection rates of the comparison algorithms and the proposed algorithm under these different dimensionality levels. It can be seen that the traditional SVM algorithm has a low detection rate for jewelry, far below that of the other four algorithms. Since the images in the self-built jewelry database closely resemble photographs taken in real life, there is significant variation in angles, facial expressions, poses, and lighting. Compared to standard databases such as Extended Yale B and ORL, the intra-class variation is substantial. The SVM algorithm exhibits very low robustness to such noise and can correctly detect only about half of the jewelry images in the test set, whereas the detection rates of MLC and KNN show a significant improvement compared to SVM. In contrast, the LRR-SRC algorithm incorporates sparse and low-rank matrix decomposition to recover the inherently low-rank features from noisy jewelry images, thereby reducing the noise interference that affects similarity. As a result, the algorithm's detection rate outperforms those of SVM, MLC, and KNN.

***Table 4.*** Detection rates of several methods on the self-built jewelry dataset

| Feature dimension | SVM | MLC | KNN | LRR-SRC | This paper |
|---|---|---|---|---|---|
| 20 | 50.28% | 59.36% | 69.77% | 80.05% | 84.83% |
| 50 | 55.96% | 62.08% | 73.08% | 81.33% | 85.26% |
| 100 | 56.44% | 65.73% | 76.28% | 84.72% | 86.95% |
| 200 | 56.94% | 66.19% | 77.73% | 85.28% | 87.40% |
| 400 | 57.28% | 66.72% | 78.05% | 85.86% | 88.09% |

The algorithm described in this paper incorporates the transformation matrix of the aligned images into the objective function of the low-rank matrix recovery model. Additionally, the LADMAP mechanism is integrated into the constraint terms for the low-rank matrix to reduce its intra-class variance while increasing its inter-class variance. These methods enable the training sample images in the unconstrained jewelry database to better filter out noise and more effectively capture the features of the jewelry. The incorporation of LADMAP also results in a detection rate for the proposed algorithm that is slightly higher than that of algorithms without LADMAP.

Furthermore, the OMP extended dictionary constructed using both low-rank and sparse components is fully capable of describing jewelry images containing significant errors. It provides an excellent training dictionary for the classification recognition of sparse representation classifiers, resulting in a detection rate for the proposed algorithm that is significantly higher than that of the conventional LRR-SRC algorithm, which combines low-rank matrix recovery with sparse representation classification. Experimental results demonstrate the effectiveness of the proposed algorithm in performing detection tasks on the self-built jewelry database.

## 4. Conclusion

Accessory detection is a challenging research topic with great potential. Due to factors such as the external environment, occlusion, and shooting methods affecting the acquisition of images of people wearing accessories, the actual accessory images obtained rarely exhibit low-rank structures, which adversely affects classification performance. To address this issue, this paper proposes an accessory detection method based on low-rank matrix recovery and sparse representation. By incorporating a regularization term into the objective function of low-rank matrix recovery, category label information is integrated into the dictionary learning process, enabling the learning of a discriminative dictionary for performing low-rank matrix recovery on training samples. Simultaneously, a low-rank projection matrix is learned to project test samples into the corresponding underlying subspace, thereby eliminating errors in the test samples. Finally, classification is performed using a sparse representation-based classification method. Experimental results show that in ablation studies, the jewelry detection algorithm—which integrates sparse representation (SRC), low-rank matrix recovery (LRR), adaptive penalty linearized alternating direction method (LADMAP), and OMP algorithm optimization—achieved an accuracy of 0.9761 and an average precision of 83.71%, representing improvements of 0.1507 and 16%, respectively, over the baseline model. In comparative experiments, the detection rates of our method across 20 to 400 dimensions were significantly superior to those of traditional classification algorithms such as SVM, MLC, and KNN. In future work, we will further optimize the model architecture and parameters, and expand our custom jewelry dataset to include more data from complex environments and a wider variety of items, in order to enhance the model's robustness and generalization capabilities.

**References**


1. Artiukh, T., Hryhorenko, I., Ternova, A., Yaheliuk, S., Verenikin, O., & Cernavca, M. (2021). Identification of white jewelry alloy based on silver and platinum for testing purposes. Eastern-european journal of enterprise technologies, 47-59.
2. Mueanglue, P., & Chompu-inwai, R. (2021, September). Defective product reduction in jewelry part production using six sigma technique. In AIP Conference Proceedings (Vol. 2397, No. 1, p. 020006). AIP Publishing LLC.
3. Zhang, Y., Fan, Y., Xu, M., Li, W., Zhang, G., Liu, L., & Yu, D. (2020). An improved low rank and sparse matrix decomposition-based anomaly target detection algorithm for hyperspectral imagery. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 13, 2663-2672.
4. Sun, W., Liu, C., Li, J., Lai, Y. M., & Li, W. (2014). Low-rank and sparse matrix decomposition-based anomaly detection for hyperspectral imagery. Journal of Applied Remote Sensing, 8(1), 083641-083641.
5. Peng, H., Li, B., Ji, R., Hu, W., Xiong, W., & Lang, C. (2013, June). Salient object detection via low-rank and structured sparse matrix decomposition. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 27, No. 1, pp. 796-802).
6. Zhang, Z., Xu, Y., Yang, J., Li, X., & Zhang, D. (2015). A survey of sparse representation: algorithms and applications. IEEE access, 3, 490-530.
7. Wohlberg, B. (2015). Efficient algorithms for convolutional sparse representations. IEEE Transactions on Image Processing, 25(1), 301-315.
8. Tong, L., Wong, W. K., & Kwong, C. K. (2017). Fabric defect detection for apparel industry: a nonlocal sparse representation approach. IEEE access, 5, 5947-5964.

9. Chandrasekaran, V., Sanghavi, S., Parrilo, P. A., & Willsky, A. S. (2009). Sparse and low-rank matrix decompositions. IFAC Proceedings Volumes, 42(10), 1493-1498.
10. Nguyen, L. T., Kim, J., & Shim, B. (2019). Low-rank matrix completion: A contemporary survey. IEEE Access, 7, 94215-94237.

**About the Author**
**Xi Jun** (b. 1984), Ph.D., is a researcher at Hohhot Minzu College, Inner Mongolia, China. His primary research direction covers arts and crafts, mainly focusing on the inheritance, development and innovative application of traditional and modern arts and crafts, with persistent academic research in related fields.