Frontiers of Information Technology & Electronic Engineering
www.jzus.zju.edu.cn; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

Visual interactive image clustering: a target-independent approach for configuration optimization in machine vision measurement*#

Lvhan PAN†, Guodao SUN†‡, Baofeng CHANG, Wang XIA,
Qi JIANG, Jingwei TANG, Ronghua LIANG

College of Computer Science and Technology, Zhejiang University of Technology, Hangzhou 310012, China

†E-mail: lvhanpan@zjut.edu.cn; guodao@zjut.edu.cn

Received Nov. 5, 2022; Revision accepted Jan. 5, 2023; Crosschecked Feb. 2, 2023

Abstract: Machine vision measurement (MVM) is an essential approach that measures the area or length of a target efficiently and non-destructively for product quality control. The result of MVM is determined by its configuration, especially the lighting scheme design in image acquisition and the algorithmic parameter optimization in image processing. In a traditional workflow, engineers constantly adjust and verify the configuration for an acceptable result, which is time-consuming and significantly depends on expertise. To address these challenges, we propose a target-independent approach, visual interactive image clustering, which facilitates configuration optimization by grouping images into different clusters to suggest lighting schemes with common parameters. Our approach has four steps: data preparation, data sampling, data processing, and visual analysis with our visualization system. During preparation, engineers design several candidate lighting schemes to acquire images and develop an algorithm to process images. Our approach samples engineer-defined parameters for each image and obtains results by executing the algorithm. The core of data processing is the explainable measurement of the relationships among images using the algorithmic parameters. Based on the image relationships, we develop VMExplorer, a visual analytics system that assists engineers in grouping images into clusters and exploring parameters. Finally, engineers can determine an appropriate lighting scheme with robust parameter combinations. To demonstrate the effectiveness and usability of our approach, we conduct a case study with engineers and obtain feedback from expert interviews.

Key words: Machine vision measurement; Lighting scheme design; Parameter optimization; Visual interactive image clustering

https://doi.org/10.1631/FITEE.2200547 **CLC number:** TP39

1 Introduction

Product quality control is the last but necessary production phase in industry, especially in manufacturing and process industries, to ensure that a manufactured product adheres to several quality specifications or meets customers' requirements (Albers et al., 2016; Godina and Matias, 2019). Machine vision measurement (MVM) (Alonso et al., 2019), as a crucial approach in product quality control, has been widely used to measure the area or length of a target to guarantee dimensional specifications in a non-destructive, low-cost, and efficient manner. Because hardware cost has been reduced, more small factories are replacing inefficient and expensive labor with

‡ Corresponding author

* Project supported by the National Key R&D Program of China (No. 2020YFB1707700), the Zhejiang Provincial Natural Science Foundation of China (No. LR23F020003), and the National Natural Science Foundation of China (Nos. 61972356 and 62036009)

Electronic supplementary materials: The online version of this article (https://doi.org/10.1631/FITEE.2200547) contains supplementary materials, which are available to authorized users

ORCID: Lvhan PAN, https://orcid.org/0000-0002-8272-6096; Guodao SUN, https://orcid.org/0000-0002-8383-8153

MVM that automatically and accurately measures the target. When front-line engineers perform a specific measurement task, where the detection target and measurement accuracy (here, the measurement accuracy is defined as the closeness of agreement between a measured value and a true value of the target) are determined, they add lighting scheme design and algorithmic parameter optimization, that is, the configuration optimization of MVM (Martin, 2007).

The goal of lighting scheme design is to acquire a high-quality image that has clear boundaries delineating the target in contrast with the background. Although some practical guidance and research about the choice of light sources facilitate the complicated process (Martin, 2007; Yuan QM and Zhang, 2022), it is still time-consuming and tedious for engineers to design an appropriate solution. They need to determine a suitable combination from various light sources, lenses, and other hardware, considering the characteristics of the target. In addition, redesigning the lighting scheme is inevitable because of the conversion from a stable test environment to an uncertain actual environment.

After image acquisition, engineers develop an algorithm and optimize parameters to process images. One problem that restricts parameter optimization is parameter importance; e.g., engineers should preferentially adjust important parameters that significantly impact the result. The other problem is determining the subsequent parameter with its adjustment range after adjusting one parameter, that is, parameter correlation. In addition to the aforementioned problems, the scale of parameters has the most influence on the difficulty of parameter optimization.

To address these challenges in MVM lighting scheme design and parameter optimization, we closely collaborate with front-line engineers to further grasp their requirements. We propose visual interactive image clustering, a target-independent approach that relaxes two problems simultaneously. In our approach, engineers first acquire images under several candidate lighting schemes and develop an algorithm for image processing. They determine parameter ranges, sample interesting parameters, and execute the algorithms with all parameter combinations to obtain results for each image. In data processing, we propose an explainable weighted Jaccard index to calculate the similarity between parameters in measuring the relationship between two images, without laboriously choosing suitable feature descriptors. After calculating all relationships, we generate an image relationship graph where the node and edge correspond to the image and measurement respectively, and propose two complementary edge-filtering methods to obtain valuable edges. We further use spectral clustering, an effective method that groups the graph into different clusters, to determine an initial clustering result as a starting point for subsequent visualization.

To achieve good clustering results, we design VMExplorer, an interactive visualization system that supports engineers in exploring appropriate configuration of MVM with four views. Cluster view is a primary overview that displays the clustering result with multi-layer bubbles that identify and layer each cluster. In this view, each image with sampled parameters encodes a steerable glyph. Engineers can conveniently drag each glyph to change its cluster, empty or create a cluster, and merge different clusters. Intersection view, a visualization with pixel-based images and heatmaps that present multidimensional parameters at global and detail levels, supports comparison between parameters from different images. In this view, four representative metrics based on different perspectives are proposed to reveal outliers that divide a group of images selected in the cluster view into more independent clusters. The correlation view not only uses a matrix-based line chart to investigate parameter correlation and parameter importance, but also provides a superposition design to further compare parameters from different images. The detail view provides basic information, including the acquired images, the developed algorithm, and the sampled parameters with ranges. To evaluate our approach, we conduct a case study with front-line engineers who have different levels of expertise and experience. We hold interviews with them to gather feedback, and discuss the strengths and limitations of our approach.

In summary, the main contributions of this paper include the following: (1) We propose an explainable measurement to assess the relationships among images using algorithmic parameters; (2) We propose a target-independent approach, visual interactive image clustering, which is based on parameter sampling and visualization to facilitate configuration optimization in MVM; (3) We present VMExplorer (https://github.com/6sixteen/VMPE), a visual analytics system with coordinated views, which supports the configuration exploration of images and parameters.

2 Related works

In this section we look at related works from three perspectives: MVM system (MVMS) which introduces representative measurement systems and research concerning lighting scheme design and parameter optimization, visual parameter space analysis (VPSA) which provides a general framework for our approach, and static set visualization which inspires our visual design.

2.1 Machine vision measurement system

Golnabi and Asadpour (2007) first discussed system design methodology and reported a generic model to guide the design of an MVMS. Various MVMSs have been developed to measure different targets with automatic machines (Luk et al., 1989; Wang et al., 2015; Ngo et al., 2017). To mitigate inefficiencies in traditional methods that detect gears, Wang et al. (2015) proposed a novel method for gear parameter measurement based on computer vision. Experimental results showed that the system was stable and fast to replace manual detection in actual production. Ngo et al. (2017) built a three-dimensional (3D) measurement system to measure in-process mechanical parts, and verified it with more precise measurement equipment to demonstrate the usability of their system.

In any MVMS, low-level image processing is an essential first step that requires good lighting to obtain high-quality images. High-quality images can greatly reduce the difficulty of algorithm development and cost of the system. Lighting scheme design focuses on the type, intensity, number, layout, and color of the light sources (Yuan QM and Zhang, 2022). Kopparapu (2006) suggested a design procedure for determining the optimal position of the light sources to obtain uniform illumination. Martin (2007) provided a practical guide to machine vision lighting. When designing an appropriate lighting scheme, engineers must consider the characteristics of the targets. Zorcolo et al. (2011) investigated the factors that mostly affect the accuracy in a typical measurement application. However, our approach is target-independent, which means that it is applicable to most targets without considering their characteristics. The underlying principle of this feature is that we propose an explainable measurement to calculate the relationships among images under different lighting schemes with algorithmic parameters that are directly related to the results.

After designing an appropriate lighting scheme, the last part of an MVMS is the algorithm development. An algorithm is a pipeline that consists of some vision operations arranged in a certain order. An operation is a particular and well-defined function with several parameters, performing specific actions on its input data in the form of images from any operation preceding it. Operations, such as thresholding operations (Sahoo et al., 1988) and morphological operations (Comer and Delp, 1999), are frequently used in MVM due to the simple principle and significant effect. The output of the intermediate operation is difficult to perceive and compare, because the output presents the feature of the target, which hinders parameter optimization.

2.2 Visual parameter space analysis

VPSA is divided into sampling and visualization. Interesting parts of the parameter space are sampled, and the outputs of all samples are computed. Visual methods are created to explore the parameter space while keeping human in the loop.

Interactive visual approaches to parameter space analysis have been successfully applied in different areas, including image processing (Pretorius et al., 2011, 2015; Torsney-Weir et al., 2011), expansion of diseases (Afzal et al., 2011), multidimensional simulation models (Bergner et al., 2013), animations (Bruckner and Möller, 2010), and 3D geometry (Coffey et al., 2013). In image analysis, Torsney-Weir et al. (2011) proposed Tuner, where the input-output models are brain segmentation algorithms. They applied a statistical model to estimate the response of the segmentation algorithm and guided the user toward better results by establishing additional sample points with 12 objective measures. The difference between our work and theirs is that they optimized parameters only in a specific image, while our approach deals with a set of images simultaneously and finds corresponding robust parameters.

To handle high-dimensional sampled data, dimensionality reduction (DR) techniques are widely used to generate two-dimensional (2D) projections and explore cluster structures of the original data (Xia et al., 2018, 2020, 2022, 2023). Xia et al. (2022) presented the results of a user study that investigates the influence of different DR techniques on visual cluster analysis (cluster identification, membership identification, distance comparison, and density comparison). In addition to the above frequently used DR techniques (such as t-SNE and UMAP), interactive DR techniques have been proposed to support users interactively steer the DR models in semi-automatic settings. For example, Xia et al. (2023) proposed an interactive DR approach with contrastive learning for interactive visual cluster analysis.

Many successful works not only show the great potential of VPSA in using and validating simulation models, but also express the common structure and tasks. Sedlmair et al. (2014) provided a conceptual framework with three major components based on their own experience and a structured analysis of the visualization literature. This framework guides and systematizes our approach, and we further broaden the limits of the almost unchanged environmental variable (images) generalized in the framework to a controllable variable according to our tasks and requirements.

2.3 Static set visualization

We consider the parameter combinations of each image as a static set, so we introduce works about static set visualization where the elements of sets remain unchanged over time.

The major challenge in static set visualization is the scalability problem (Zhu et al., 2022). The number of relationships between sets explodes exponentially when sets expand. For an overview of state-of-the-art techniques of set visualization, we recommend the survey by Alsallakh et al. (2016). They classified visualization techniques into seven categories according to the visual representations and tasks. Euler diagrams are among the oldest and most popular set visualizations (Baron, 1969). Sets and set relations are represented by labeled closed curves and overlapping curves, respectively.

Some researchers used matrix-based visualization to relax the scalability problem. For instance, the OnSet scheme uses a matrix representation to encode large-scale binary set data to matrices (Sadana et al., 2014). UpSet (Lex et al., 2014), another successful work, uses a matrix layout where each row represents an overlap to analyze a specific overlap quickly. Inspired by these works, we propose a matrix-based visualization with pixel-based images and heatmaps to present parameters at global and detail levels. Set intersections occupy a large part of our approach and there are some valuable approaches for visualizing set intersections (Yalcin et al., 2016; Alsallakh and Ren, 2017). AggreSet (Yalcin et al., 2016) aggregates attributes through circle glyphs to show set intersections in the matrix. It supports comparison as core exploratory tasks and allows analysis of set relations. PowerSet (Alsallakh and Ren, 2017) shows set intersections based on treemap visualization to give an overview of nonempty intersections in the system. It enables insight into how elements are distributed across these intersections and fine-grains analysis to explore and compare their attributes in overview and detail.

3 Background

In this section we first describe the typical workflow in MVM and indicate the time-consuming parts that require considerable expertise and extensive practical experience. We then describe four specific tasks derived from front-line engineers through close collaboration.

3.1 Machine vision measurement

MVM is frequently applied to measure the size of a target with an automatic measurement system (Ngo et al., 2017). In a specific measurement task, engineers design and develop a complete solution, where they struggle to design an appropriate lighting scheme to acquire high-quality images and optimize parameters to obtain acceptable measurement results.

The lighting scheme design includes the choice of the light source, lens, camera, and manipulator (for feeding), as well as the placement of these items (Golnabi and Asadpour, 2007). To acquire high-quality images, engineers adjust the lighting scheme skillfully and iteratively to make the boundaries of targets and backgrounds clear. The characteristics of the target surface determine the difficulty of lighting scheme design. Some parts of the target may reflect more light than others, which causes the image to map the actual target boundaries imprecisely (Martin, 2007). A suitable light source can prevent the camera from receiving too much or too little light, and create high contrast between the target boundaries and the background. However, the actual environment is more complex and changeable than a test environment, which makes the original design invalid in an actual environment.

After designing a lighting scheme, engineers acquire test images with different placements of the target to perform adequate testing. They spend an abundance of time developing an algorithm and optimizing parameters. We ignore the development of the algorithm and pay attention to parameter optimization that is considerably common and essential in MVM. Parameter optimization is composed of three parts: an engineer (1) runs the algorithm with specific parameters, (2) inspects the output, and (3) re-runs the algorithm with refined parameters if the output is unsatisfactory. It is inevitable that elaborately optimized parameters are suitable for one image, but invalid for another. To find common parameters for all images, engineers may return to the previous steps, such as modifying the lighting scheme, or even operating the manipulator to maintain consistency in target placements. In conclusion, parameter optimization is a tedious trial-and-error procedure as a result of parameter combinations and interruptions of algorithm executions.

3.2 Task analysis

We derive four specific tasks to help design lighting schemes and optimize parameters during a long-standing continued partnership with engineers. We initially conduct a questionnaire survey with 25 engineers to gain preliminary insight. Eleven engineers have $\geq$2 years of work experience (WEX). Eight engineers have half a year to two years of WEX, and the remainder are inexperienced in MVM, with $\leq$3 months of WEX. We choose two engineers for further collaboration. Engineer A (E_A) has been employed in this industry for three years, measuring nuts and screws in his daily work. Engineer B (E_B), with six months of WEX, switched his career from a front-end engineer to an apprentice who designs lighting schemes. Based on our analysis of the questionnaire and discussion with these two engineers (E_A and E_B), four significant tasks are summarized as follows:

T1: design an appropriate lighting scheme

An appropriate lighting scheme is the key to acquiring high-quality images, which is extremely important in determining whether the measurement task is successful or not. However, it is an intricate process to quantify the influence of changes in the lighting scheme on the measurement result, particularly, if the target has a complex structure that interferes with the measurement. Although engineers summarize some tips for solving common problems, they still have to explore an enormous search space. The hardware budget is another constraint in actual production, and requires engineers to design a lighting scheme at a less cost.

T2: optimize algorithmic parameters

The purpose of algorithmic parameter optimization is to find common and robust parameters for images where the placements of the targets are diverse. To achieve this purpose, engineers repeatedly choose possible parameters, run the algorithm, and observe the results, or even modify the lighting scheme. This inefficient and tedious approach is inadequate due to the limitation of time cost. Specifically, E_B mentioned that he usually attempted to seek non-existent common parameters for two visually similar images that actually had unique parameters, which resulted in a waste of time and energy.

T3: understand parameter importance

In the questionnaire, engineers pointed out that they struggled to distinguish the importance of a parameter, especially for unfamiliar operations where they typically used default recommended values. Because major parameters significantly impact the result, whereas minor parameters impact the result only slightly, it is valuable to identify major parameters and concentrate efforts on adjusting them. Another essential reason for understanding parameter importance is from the clients' perspective. They are generally inexperienced factory workers or individual users with limited knowledge about machine vision or MVM. They need a few major parameters exposed in the final graphical interface software to adjust, so they can still finish their measurement tasks in the complex actual environment.

T4: understand parameter correlation

This challenge, coming mainly from novices, is that they aspire to exclude invalid attempts at parameter optimization by understanding parameter correlation.

4 Approach

In this section, we first introduce the pipeline of our approach, and then introduce each stage of the approach in turn. The visualization system, which is another key point, is introduced in Section 5.

4.1 Pipeline

To solve tasks derived from engineers, we propose a target-independent approach, visual interactive image clustering, which facilitates lighting scheme design and algorithmic parameter optimization and simultaneously supports the exploration of parameter importance and parameter correlation. Our approach interactively groups images acquired under various lighting schemes into different clusters using parameters, and suggests a lighting scheme with robust parameters for each cluster. Our approach contains four steps: data preparation (Fig. 1a) including image acquisition and algorithm development, data sampling (Fig. 1b) to obtain parameter sets with results for each image, data processing (Fig. 1c) to obtain a filtered relationship graph and an initial clustering result, and visual analysis with our visualization system (Fig. 1d) to comprehend parameters and facilitate subsequent clustering.

4.2 Data preparation

During preparation, engineers design several candidate lighting schemes by observing the acquired images directly and then change the placement of the target to obtain test images for all lighting schemes. For algorithm development, we develop graphical software that is based on Halcon (professional software for machine vision with an integrated development environment) to help engineers develop a feasible algorithm swiftly by dragging vision operation modules without coding. The algorithm is verified only in partial images and partial parameters.

4.3 Data sampling

After preparing for alternative lighting schemes and a feasible algorithm, engineers continue to use our graphical software to sample data conveniently. They determine critical parameters and select the parameter ranges, including the minimum, maximum, and step size. For each image, the software automatically executes the algorithm with all parameter combinations to obtain and restore results. For all parameter combinations, unsampled parameters remain at the same default values.

Here is a concrete example to introduce data sampling: As shown in Fig. 2a, engineers want to measure the area of the white pin of the complex printed circuit board (PCB) which is identified by the blue mask. After the lighting design

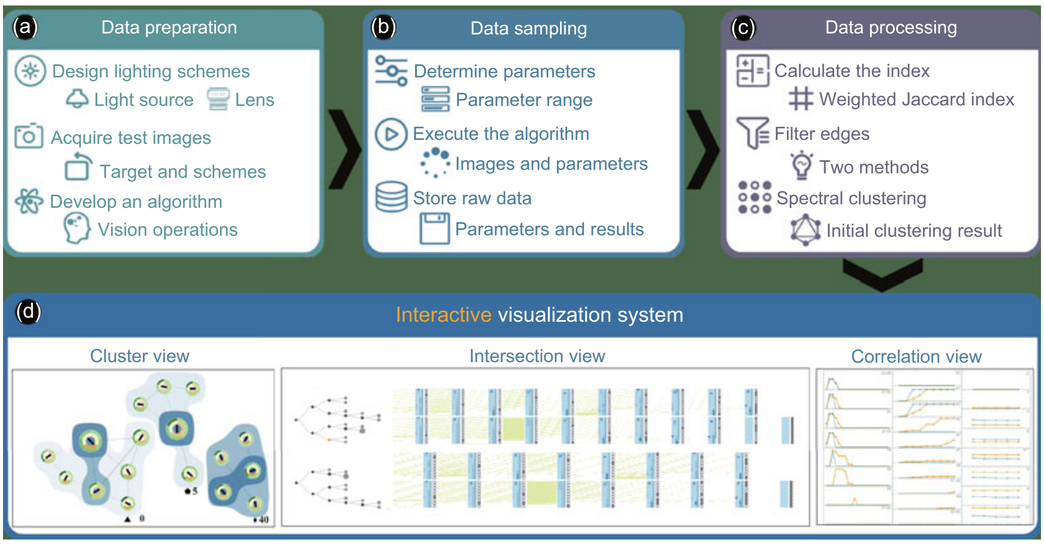

Fig. 1 Pipeline of visual interactive image clustering: (a) data preparation; (b) data sampling; (c) data processing; (d) visual analysis with our visualization system

and algorithm development period, engineers select three essential operations for sampling, including thresholding which segments an image using a global threshold, opening_circle which opens a region with circular structuring, and select_shape which chooses regions with the aid of shape features. Sampled parameters with ranges are illustrated in Table 1; for example, the first row in Table 1 indicates that the value of minThreshold from thresholding can be 20, 21, 22, $\cdots$, 40. We define a combination of certain values of all sampled parameters as a parameter element; for instance, 20 (minThreshold), 40 (maxThreshold), 2 (openingCircle), and 0.84 (minShape) constitute a parameter element. The algorithm result includes a numeric value for each parameter element, and for an image, all parameter elements form a parameter set. We illustrate these concepts by mathematical expressions in the supplementary materials for generalization.

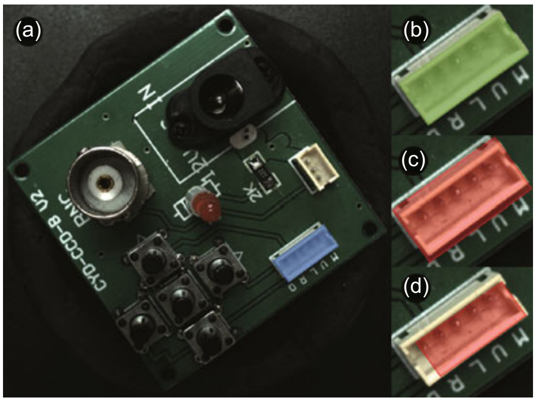

Fig. 2 Different measurement results for a same image: (a) the target is the area of the white pin of the printed circuit board (blue mask); (b) appropriate parameters produce an acceptable measurement result that satisfies the measurement accuracy requirement (green mask); (c) inappropriate parameters produce an unacceptable measurement result that contains an additional non-target area (big red mask); (d) inappropriate parameters produce an unacceptable measurement result that lacks the necessary target area (small red mask) (References to color refer to the online version of this figure)

Table 1 Parameters with ranges in data sampling

Parameter	Operation	Min	Max	Step size
minThreshold	thresholding	20	40	1
maxThreshold	thresholding	40	60	1
openingCircle	opening_circle	2	10	1
minShape	select_shape	0.8	0.84	0.01

Min for minimum and max for maximum

4.4 Data processing

Data processing is described in four parts: fundamental concepts, the weighted Jaccard index, two complementary methods to filter edges of the image relationship graph, and spectral clustering to cluster the filtered graph.

1. Fundamental concepts

The parameter elements of a parameter set can be divided into success and failure according to whether the algorithm can be completely executed to obtain a result. The reason for failure is that the preceding operation with inappropriate parameters produces an empty output for the succeeding operation. Success can be divided further into two categories: the filtered parameter set whose results satisfy the measurement accuracy requirement and the dirty parameter set whose results violate the accuracy. The acceptable result is indicated by the green mask in Fig. 2b, while the unacceptable result is indicated by the big red mask, which contains a non-target area (Fig. 2c), or the small red mask, which lacks the target area (Fig. 2d). We provide a figure in the supplementary materials to illustrate these concepts clearly.

Intersection and union of filtered parameter sets are frequently used in data processing. Filtered parameter sets can intersect mutually to obtain a new set, which is composed of parameter elements that are common to both sets. The union of two filtered parameter sets generates a new set containing all parameter elements in both sets. The intersection of more than two filtered parameter sets is the repetition of the operation with two sets until a final intersected set is generated, and the union is a similar process. Although the final set is invariable if all original sets are determined, the intermediate sets are different with different selection of two sets in each step, that is, intersection order.

2. Weighted Jaccard index

The Jaccard index is widely applied to calculate the similarity between two sets in domains where binary data are used. We consider the filtered parameter set as the set for calculating the Jaccard index, and ignore the numerical results of the set in the calculation because they already satisfy the accuracy requirement. This measurement is explainable when measuring the relationship between two images. For example, a value of 0 indicates that the images vary widely, while a value closer to 1 means that the images are more similar. However, this measurement ignores the frequency of the filtered parameter elements in all filtered parameter sets, which is the ratio of the number of the elements that appear in all sets to the number of all sets. The high-frequency element indicates that most images can have acceptable results with the element that should receive greater attention. Considering this feature, we propose a weighted Jaccard index in Eq. (1), where each element is multiplied by its frequency f and $\boldsymbol{P}$ is the filtered parameter set. Actually, the Jaccard index is a particular case of the weighted Jaccard index, where each set has the same frequency, which equals 1.

$$\mathrm{WJI}\left(\boldsymbol{P}_i, \boldsymbol{P}_j\right)=\frac{\sum\limits_{e} f^{e}}{\sum\limits_{e'} f^{e'}}, \tag{1}$$

where

$$\begin{cases} e \in \boldsymbol{P}_i \cap \boldsymbol{P}_j, \\ e' \in \boldsymbol{P}_i \cup \boldsymbol{P}_j. \end{cases} \tag{2}$$

3. Edge filtering

An image relationship graph is constructed after calculating the weighted Jaccard index for all filtered parameter sets. The filtered parameter set of an image is represented as a node in the graph, and each relationship between two images is an edge connecting two corresponding nodes. To understand the image relationship using the graph, we propose two complementary methods for extracting valuable edges, which also reduce visual clutter in the visual analysis.

The concept of method 1 is to extract several essential edges with large indices only. Taking an image as an example, method 1 first calculates the weighted Jaccard index with other images to obtain a vector, and then sorts the vectors in descending order. Finally, the top p ($p=30\%$ in our experiments) edges are preserved. However, method 1 may ignore significant edges when small filtered parameter sets produce similar indices, because a large denominator plays a decisive role in calculating the index. To complement method 1, we propose method 2, which applies the size of the intersection of two parameter sets as the calculation criterion. In addition to the calculation criterion, the pipeline of method 2 is identical to that of method 1. The final edges are the combination of edges generated by these two methods.

4. Spectral clustering

After obtaining the filtered graph, we apply the spectral clustering algorithm with excellent clustering effect to cluster the graph initially. The initial clustering result is generally an undesirable state, where some clusters still have common filtered parameter elements. However, our approach requires independent clusters to suggest a different lighting scheme with robust parameters for each cluster. For better configuration, clusters should have no common parameters that produce acceptable results, and the number of clusters should be as small as possible. The former constraint guarantees cluster independence, and the latter improves the differentiation.

Visualization is a tool for resolving this phenomenon with intuitive presentations of multi-dimensional data (Yuan J et al., 2021). Thus, we design our visualization system to obtain better clustering results.

5 VMExplorer

In this section, we first describe the design requirements of our visualization system (VMExplorer) which are guided by the tasks in Section 3.2, and then describe the system design.

5.1 Design requirements

R1: provide an overview equipped with necessary interactions to explore clusters

The system needs an overview to present clusters of all images and provide necessary interactions to get engineers into the subsequent clustering loop (T1). The success that makes the algorithm be executed completely should be visualized as the primary object in the overview to represent the corresponding image and its lighting scheme (T2). Engineers can observe the distribution of the parameter set and compare different parameter sets. The relationship between two images calculated by the weighted Jaccard index is supposed to be provided to engineers to discover special patterns, such as two irrelevant images without a common parameter element (T1). Parameter sets in the same cluster are expected to be located in adjacent areas, and each cluster needs an obvious boundary as identification (T1). For interactions, engineers can view the original images and arbitrarily move each image from one cluster to another to change its class. The system also allows engineers to create a new cluster, empty an original cluster, and select a group of objects to observe their properties for better clustering (T1, T2).

R2: support a scalable view that presents filtered parameter sets to facilitate clustering

To assist engineers in grouping images with latent similar properties into the same cluster, the system should encode a filtered parameter set in an identifiable way and provide a scalable view to present a group of filtered parameter sets (T1, T2, T3). First, the filtered parameter set is multi-dimensional, and its size explodes exponentially with an increase in the quantity and ranges of parameters. Second, the number of filtered parameter sets is unrestricted, determined entirely by engineers.

R3: recommend significant intersection orders to identify special images in a cluster

In addition to supporting a scalable view to present filtered parameter sets, the system should suggest meaningful intersection orders to facilitate seeking special images (T1, T2). A special image with an unusual filtered parameter set considerably influences the final intersected filtered parameter set in a cluster, so it is critical to exclude it from the cluster to force the cluster to have more common parameter elements (T1, T2).

R4: describe filtered parameter sets from multiple perspectives to understand parameters

Because a filtered parameter set with vast data volume can be represented as an enormous matrix, the matrix cannot be observed directly to understand parameter importance and parameter correlation. Thus, the system needs multiple perspectives for presenting filtered parameter sets. Statistical perspectives, such as parameter frequencies, omit partial information, but emphasize a particular aspect of the filtered parameter set (T3). To understand parameter correlation, instead of showing parameters in table form, a scalable and intuitive visualization should be designed, considering that engineers are unfamiliar with visualization (T4).

5.2 System interface

Under the guidance of design requirements, we design VMExplorer (Fig. 3), a visualization system that supports subsequent image clustering for engineers to optimize configuration of MVM. In this subsection, we introduce the visual encodings and interactions of each view.

5.2.1 Detail view

Detail view (Fig. 3a), which provides fundamental information including the algorithm, images, and sampled parameters with defined ranges, allows engineers to have a global grasp of the explored target. Engineers can drag the scroll bar to observe the images on the left side and determine the initial number of clusters based on experience for spectral clustering. Considering that inexperienced engineers struggle to assign the initial number, we use a perceptual hash algorithm (Weng and Preneel, 2011) to extract the image features, calculate the similarity between two images by the European distance, and employ mean shift clustering (Wu and Yang, 2007) to recommend the number of clusters. On the right side, the algorithm flow chart is illustrated with the sampled parameters highlighted in green and the remainder in black, where engineers can monitor the minimum, maximum, and step size of the sampled parameters. To save system space, we add a sidebar which can be popped and hidden from the far right of the system interface by pressing X. In the sidebar, engineers can enter system parameters, such as the number of initial clusters and the range of measurement accuracy.

5.2.2 Cluster view

Cluster view (Fig. 3b) contains three major components: the glyph representing the image, the edge describing the relationship between two images, and the bubble identifying the clusters.

For glyph design, the glyph area encodes the size of the filtered parameter set, because the area is an expressive visual channel to represent ordered attributes (Munzner, 2014). We represent the distribution of the filtered parameter set and dirty parameter set with the green outer ring and orange inner ring, respectively. Because the results of the filtered parameter set are continuous, we equally divide the results into several portions and encode each portion as the part of the outer ring with the same area. These parts are arranged in clockwise order with increasing results. The number of divisions can be entered in the sidebar (5 as default). As for the dirty parameter set, we divide the inner ring into three parts. The left part indicates the parameter elements whose results are smaller than the minimum accuracy within a settable threshold, while the right indicates the larger

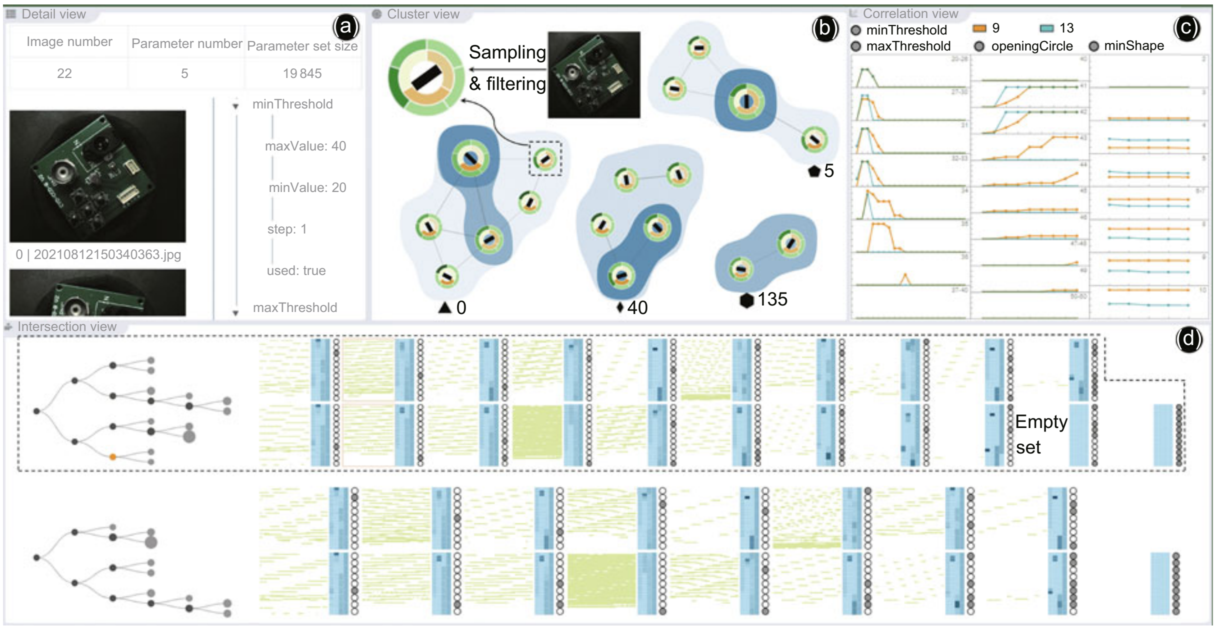

Fig. 3 System interface: (a) detail view provides basic information about the sampled data; (b) cluster view shows the clustering result with multi-layer bubbles as identification; (c) correlation view uses a matrix-based line chart to present the correlation between two parameters; (d) intersection view presents the intersection process of selected parameter sets (References to color refer to the online version of this figure)

part and the bottom describes the remainder. The number of parameter elements in each part encodes color saturation, where the darker color indicates the greater value.

The phenomenon that an image has an empty filtered parameter set implies that the image does not satisfy the measurement accuracy requirement with any currently sampled parameter set. We put the special glyph in an independent cluster (invalid cluster) for potential exploration. These glyphs may be moved from an invalid cluster when engineers extend the measurement accuracy. We use bubbles with different symbols, such as the triangle in Fig. 3b, to distinguish clusters. The size of the final intersected parameter set of the cluster is displayed on the right of the symbol. Engineers can drag any glyph from one bubble to another to change its cluster. To make the cluster organization more structured, we choose the size of the filtered parameter set as the criterion for further divisions. As shown in Fig. 3b (triangle), we select two values (50%, 75%) to divide the cluster (specifically, 75% indicates that all sets whose sizes exceed 75% of the size of the largest set are grouped in a new distinct bubble). The similarity of two filtered parameter sets, which is calculated by the explainable weighted Jaccard index, encodes the width of the edge between two glyphs. To relax visual clutter, we propose two complementary methods to filter the edges. The experimental results in the supplementary materials demonstrate that these methods can distill valuable edges.

This paragraph describes the remaining cluster view information. A black rectangle whose angle corresponds to the angle of the placement of the target is shown in the glyph. A unique identity and the original image can be checked by clicking the rectangle. The layout for all clusters is based on the force-directed graph drawing algorithm (Eades, 1984) to avoid overlap. A red curve will connect two clusters if they have common parameter sets. For interactions, engineers can merge two clusters by dragging symbols together and produce a new cluster by dragging a glyph to a blank space. Engineers can select interesting glyphs in the cluster view and monitor their intersection process in the intersection view.

5.2.3 Intersection view

Intersection view uses pixel-based images and heatmaps to illustrate the intersection process of the filtered parameter sets. Significant intersection orders are suggested for identifying outliers (i.e., special images) that have exceptional parameter distributions and considerably influence the final results.

The unique characteristic of the filtered parameter set is not well-suited for widely adopted multidimensional visualization techniques. For example, parallel coordinates cause extensive overlap of visual elements with low space utilization, because parameters are sampled in discrete mode and some filtered parameter elements differ only in the individual parameters. We propose a specifically organized pixel-based image (Fig. 4a) to encode the filtered parameter set with high space utilization. The number of parameter elements is the same in each set and equals the size of the union of sets selected in the cluster view. The green area represents the elements of the original set or intermediate set, and the white area represents missing elements relative to the union. The location of the elements is determined in descending order of parameter importance as a primary metric and parameter value as a secondary metric, which makes the pixel-based image structurally comparable.

We use a heatmap to encode parameter frequencies of the set to reflect parameter importance. The frequency of a parameter value is defined as the number of occurrences of that value divided by the size of the set. As shown in Fig. 4a, each blue column corresponds to the distribution of a parameter, and each small rectangle in the column represents the frequency of the corresponding parameter value, mapped with saturation. We define the importance of a parameter as the variance of the frequencies of all parameter values, which is accepted by the engineers. The order of columns equals the order of parameters in the algorithm, and parameter values are arranged in ascending order. We use white and gray circles to represent selected and unselected images, respectively.

To seek outliers that divide the original cluster into more clusters, we recommend four significant intersection orders. We recommend only intersection orders that intersect two filtered parameter sets in each step. We propose four meaningful metrics emphasizing different aspects to suggest intersection orders: Jaccard index (O_1), weighted Jaccard index (O_2), $|\boldsymbol{P}_1| + |\boldsymbol{P}_2|$ (O_3), and $|\boldsymbol{P}_1 \cap \boldsymbol{P}_2|/\min(|\boldsymbol{P}_1|, |\boldsymbol{P}_2|)$ (O_4) ($\boldsymbol{P}_1$ and $\boldsymbol{P}_2$ are two filtered parameter sets). Each metric derives two orders depending on whether two sets with the largest (O_D) or the smallest (O_A) metric value are intersected first. All orders share a common pipeline where they calculate the metrics of all set combinations, intersect two right sets to generate an intermediate set, and repeat the above process until the final set is generated. O_1 and O_2 both group the most similar or different sets into the same cluster, which is a general and effective concept in clustering tasks. O_3 intersects parameter sets by their size because size is an important constraint that limits the final result. O_4 emphasizes the subset relationship, and engineers can choose O_4 to find redundant images in the cluster.

Intersection view encodes the intersection order as a right-to-left tree (Fig. 3d), where the gray leaf node, the black parent node, and the black root node represent the original set, the intermediate intersected set, and the final set, respectively, with the size of the set encoding the area of the circle. The orange outlier, as shown in Fig. 3d in the upper tree, corresponds to the first empty intersected parameter set (empty set in Fig. 3d), and divides the cluster selected in the cluster view into two clusters. At the bottom part of the intersection view (Fig. 3d), an intersection process of the larger cluster generated by the outlier is exhibited by the same visual design.

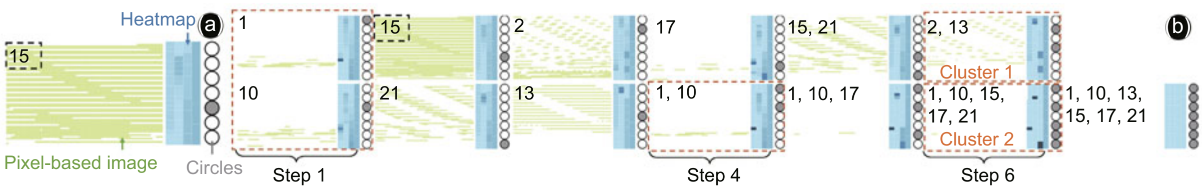

Fig. 4 Visualization of an intersection process in the intersection view: (a) each filtered parameter set encodes a combined visualization that contains a pixel-based image, a heatmap, and circles; (b) the intersection process is composed of sets selected in the cluster view and sets generated in the process (Each column represents an intersection step; for example, step 1 indicates that 1 and 10 intersect first and produce a new filtered parameter set named 1 and 10 in step 4. References to color refer to the online version of this figure)

5.2.4 Correlation view

Correlation view uses a matrix-based line chart (Fig. 3c) to support the exploration of parameter correlation of a filtered parameter set and facilitate comparison between different sets.

Correlation view visualizes only the correlation between two parameters that are in the same operation or adjacent operations. Because the traditional parameter optimization is a streamlined step-by-step process, this indicates that engineers start to tune the parameters in a new operation until the previous adjacent operation is tuned to obtain a satisfactory intermediate output. The top of the view shows parameter legends and image identities. Below the legends, the correlation between the n^{th} and $(n+1)^{\mathrm{th}}$ parameters is displayed in the n^{th} column, and the order of the parameters can be changed by clicking them. Each rectangle in the n^{th} column applies a line to display the relationship between the $(n+1)^{\mathrm{th}}$ parameter value (x-axis) and its number (y-axis), with a specified value of the n^{th} parameter marked on the right for hints (Fig. 5a). Engineers can observe the changing trend or the extreme points of the line in a rectangle, and compare the trends vertically in the direction of the column (Fig. 5b). To enhance space utilization, we merge adjacent rectangles that have the same lines to a single rectangle with a hyphen representing the range of the n^{th} parameter. For comparative visualization between two sets, we use a superposition design (Gleicher, 2018): place two lines in the same rectangle.

6 Case study: measure the area of the white pin

To evaluate the effectiveness of our target-independent approach for MVM, we invite two front-line engineers (E_A and E_B) to handle the measurement workflow with VMExplorer. Furthermore, we open our system to other engineers and conduct interviews with them to gather feedback. We then describe a case study by E_A.

1. Experimental configuration

The target is the area of the white pin of the PCB (Fig. 2a). The metal surface of the PCB tends to cause measurement interference due to the

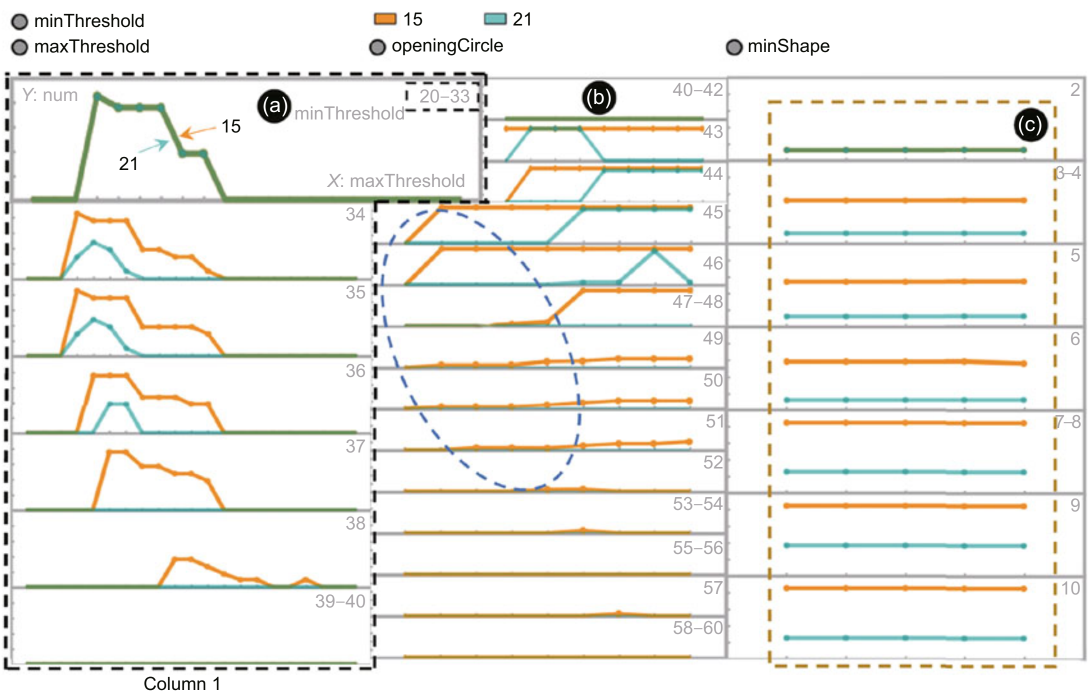

Fig. 5 A matrix-based line chart of the filtered parameter sets 15 and 21: (a) column 1 illustrates the correlation between minThreshold and maxThreshold; (b) the selected area indicates that for 15, openingCircle should be increased with the increase in maxThreshold (45–51) to obtain acceptable results; (c) a straight-line pattern of openingCircle and minShape indicates that minShape is unimportant due to the uniform distribution regardless of the values of openingCircle

light reflection. For the hardware, E_A chose a five-megapixel industrial camera (MV-CE050-30UC) produced by HIKVISION (a technology company focusing on technological innovation) and a white ring light source. E_A placed the light source directly over the PCB which was fixed on the middle of a turntable, and set three intensities of light: low, medium, and high. For each light intensity, E_A turned on the turntable and obtained images randomly with different placements. Finally, E_A acquired 22 images, each identified with a unique ID from 0 to 21, and developed a feasible algorithm swiftly with our software.

2. Data sampling and data processing

E_A selected interesting parameters, defined their ranges, and obtained all parameter sets with corresponding results. The sampled parameters were introduced in Table 1 in Section 4.3. These four representative parameters come from three typical operations, and produce a parameter set with the size of 19 845 ($21 \times 21 \times 9 \times 5$) for each image. The actual area of the white pin was 36 000 in pixels, and E_A set the acceptable measurement range from 35 000 to 37 000 to obtain the filtered parameter sets. The filtered parameter set results were ignored in data processing because they already satisfied the accuracy requirement. The filtered relationship graph was generated as the beginning for visual analysis. Before the experiment started, we introduced our system to the engineers in detail, including the visual encodings and interactions.

3. Initial stage

Initially, E_A scanned the images in the detail view to have a rough idea of the image relationship. Based on intuitive perception of image features, 3 was entered as the number of clusters. However, the cluster view presented four clusters: one cluster with an empty final intersected parameter set (failure cluster), two clusters with a non-empty set (success cluster), and an invalid cluster. In the invalid cluster, he found five glyphs (4, 5, 18, 19, and 20) whose outer rings were white. By observing images, he found that the brightness in 4, 5, 18, 19, and 20 was higher than that of others. He considered that the ranges of minThreshold and maxThreshold in thresholding mismatched the lighting conditions. He hid invalid cluster and focused on other clusters.

There were two success clusters in the initial stage: the quadrangle (2, 3, 11, 12, and 13) and the pentagon (1, 10, 15, 17, and 21). In the quadrangle, the larger glyph wrapped in an inner bubble had more edges, and occupied a more central position in the cluster. He dragged 2 and 13 (the largest and second largest, respectively) from the quadrangle to the pentagon, considering that the larger glyph with more parameter elements would not affect the final intersected parameter set of other clusters. However, the size of the pentagon decreased from 5 to 0, which means that 2 and 13 differed from the images in the original pentagon. To further explore the phenomenon beyond his expectations, he dragged 2, 13, and the pentagon together, chose the weighted Jaccard index in descending order (O_2_D) as the order, and observed their intersection process in the intersection view. As shown in Fig. 4b, he found that the final intersect parameter set was empty, but the cluster could be divided into two clusters (cluster 1: 2 and 13; cluster 2: 1, 10, 15, 17, and 21) at step 6. He noticed that two clusters had different heatmaps, especially the distribution of the second (maxThreshold) and third (openingCircle) columns, which was the cause of the phenomenon. He realized that spectral clustering produced a good initial result that grouped similar images with more of the same parameter elements together.

4. Parameter importance and parameter correlation

Subsequently, he selected the pentagon and still chose O_2_D as the order. By comparing all heatmaps in the intersection view (Fig. 6), he discovered that the color distribution of the column

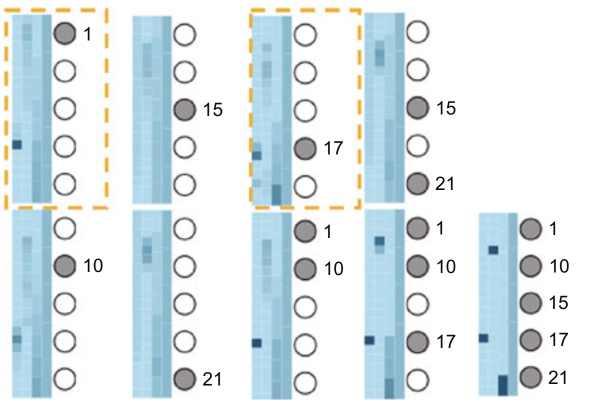

Fig. 6 Comparison of the intersection process heatmaps (1, 10, 15, 17, and 21, e.g., the pentagon) (1 and 17 are the constraints affecting the final intersected parameter set of the pentagon, with a special color distribution of the first column in the heatmap)

representing minShape was uniform, which indicates that this parameter was unimportant, because all values of minShape were sampled equally, regardless of the values of preceding parameters. He then clicked 15 and 21 in the intersection view and found that the third column describing the correlation of openingCircle and minShape in the correlation view contained only straight lines (Fig. 5c). By clicking the legends at the top of the correlation view to swap parameters, he noticed that there were still straight-line patterns, which further convinced him that the parameter was unimportant. He observed the first column (the correlation of minThreshold and maxThreshold) from top to bottom and noticed that as minThreshold increased, the line gradually shifted to the right (Fig. 5a). He realized that he should increase maxThreshold to keep the acceptable result when increasing minThreshold.

5. Interactive clustering

Finally, he performed interactive clustering to eliminate the triangle with the empty final intersected parameter set (Fig. 7a). He still chose O_2_D as the order. However, the outlier divided the original triangle into two failure clusters: the new triangle and the hexagon (Fig. 7b). Instead of continuing on this foundation, he restored the clusters to the previous stage (Fig. 7a), and attempted the Jaccard index in descending order (O_1_D) as the order, but the result was similar to that of the previous one (Fig. 7c). He realized that he should try to intersect the small parameter sets. He found that this order gave a different outlier (Fig. 7d); i.e., 0, 7, and 8 were divided into a new cluster, while a red curve appeared between the new triangle and the original quadrangle. After merging these two clusters, he successfully obtained three success clusters (Fig. 7e): the triangle, quadrangle, and pentagon. He attempted to reduce the number of clusters from 3 to 2. There were three combinations to merge these clusters, and because the pentagon was newly produced from the original triangle, he kept the pentagon as a unique cluster. He simply merged the triangle and the quadrangle to explore better clustering. Finally, he used O_4_D to intersect subsets first and found a new clustering result. That is, Glyph 16 could be moved from the triangle to the quadrangle or the opposite.

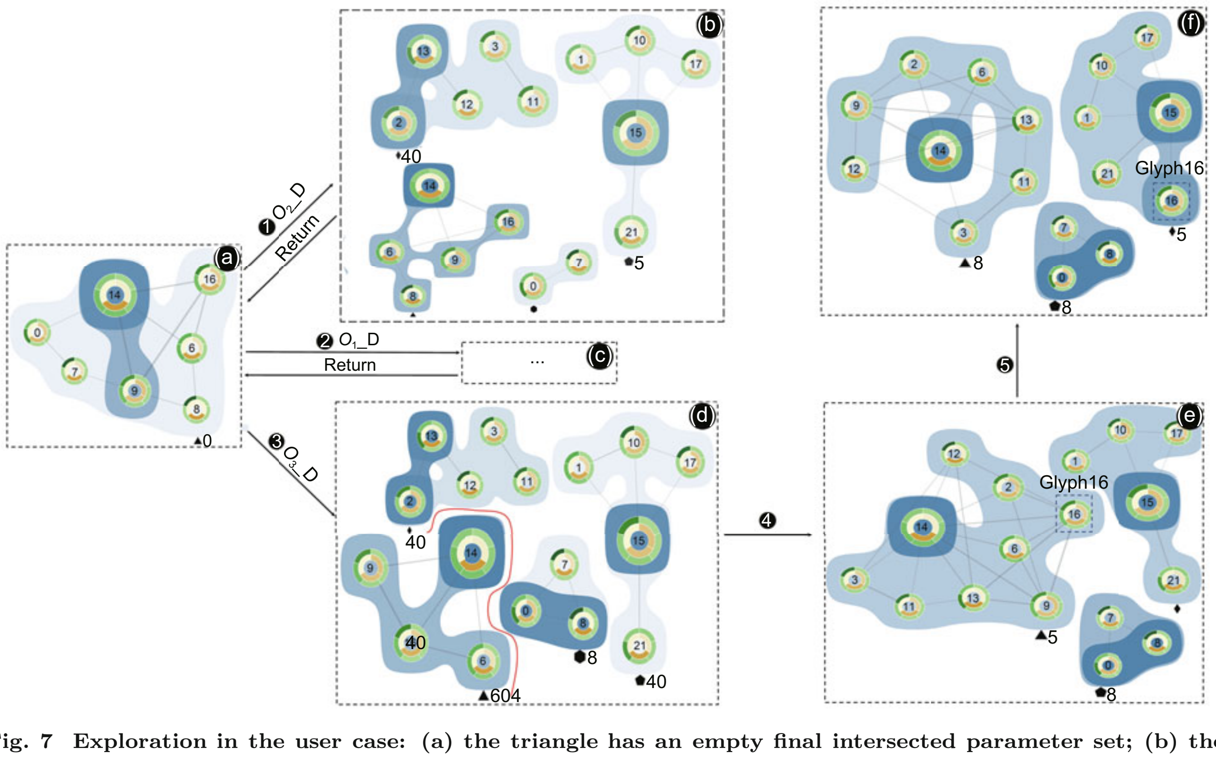

Fig. 7 Exploration in the user case: (a) the triangle has an empty final intersected parameter set; (b) the clustering result referring to O_2_D; (c) the omitted clustering result referring to O_1_D; (d) the clustering result referring to O_3_D; (e) the triangle and the quadrangle in (d) are merged to obtain a new triangle; (f) another case of three clusters (References to color refer to the online version of this figure)

6. Results and verification

E_A obtained two feasible clustering results with satisfactory parameters shown in the sidebar. Besides, he identified the unimportant parameter (minShape) and understood the correlation of parameters. He summarized that he would choose parameters of the triangle as the final parameters, because the cluster contained the most images, which means that the parameters were more robust. He chose 32, 48, 9, and 0.8 for minThreshold, maxThreshold, openingCircle, and minShape, respectively. For the lighting scheme, he chose the lighting scheme of Glyph 16 that had most edges in the triangle. Finally, he changed the placements of the target to obtain several images to verify his configuration, and the results are shown in Fig. 8, where all measurements satisfied the measurement accuracy requirement and were illustrated by the red bounding box.

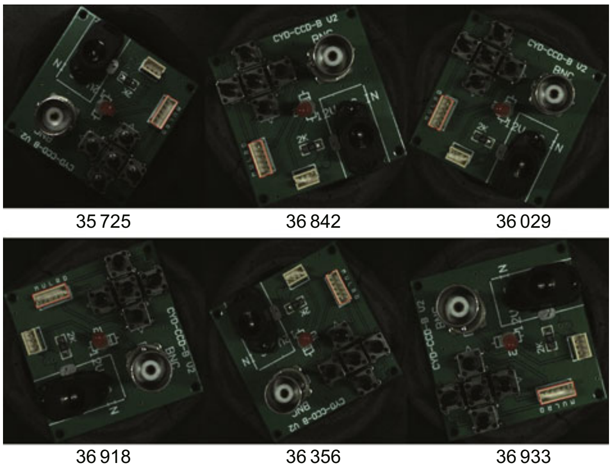

Fig. 8 Further verification of the lighting scheme with parameters of Glyph 16 (References to color refer to the online version of this figure)

7 Engineers' feedback

E_A and E_B each spent about 20 min exploring our system. We then conducted an in-depth interview with them one by one and asked them questions on usability and suggestions. In addition, our system was still available for other engineers for one day, and we received comments from five other engineers. Generally, seven engineers all agreed that our approach provides a new angle to deal with the problem of designing lighting schemes and optimizing parameters (configuration optimization). Their feedback is summarized as follows:

1. Straightforward access to image relationship

All engineers considered that measuring the relationship among images using parameters is innovative and straightforward with the advantage of ignoring the image feature. E_A stated "I usually assess the relationship between two images by comparing the image feature directly, which is entirely based on experience. This quantitative measurement that uses the relationship between parameters is explainable for me." E_B added "I can quickly find unrelated images through this system, instead of attempting various parameter combinations and waiting for the results."

2. Efficient exploration of parameters

In terms of understanding parameter importance and parameter correlation, all engineers agreed that some findings match their WEX and vision knowledge. Novices gave plenty of affirmation about the heatmap, which shows the frequency of the parameter to reflect parameter importance. One engineer mentioned "I will hide uncritical parameters found in the exploration in the software development afterward." Three engineers thought that although the matrix-based line chart is difficult to understand at first, it provides an interesting format for understanding parameter correlation by analyzing two parameter sets.

3. System design

Five engineers praised the bubbles in the cluster view, which distinguishes the clusters clearly and categorizes the glyphs in the cluster. Two engineers said "The area of the glyph is an effective visual encoding for the size of the filtered parameter set, which assists us in finding special images." Four engineers strongly praised the intersection view and our intersection orders. They emphasized that different intersection orders may lead to better clustering results in the exploration and interaction, because the choice of intersection orders is convenient. For the pixel-based image, four engineers pointed out that they can intuitively perceive the size of the filtered parameter set and find the most special one to exclude. However, three engineers said "It is difficult to compare pixel-based images of two small filtered parameter sets, and we are confused about the pattern shown by the pixel-based image." E_B denoted that he wants to observe the relationship between glyphs in different clusters in the overview. However, there are no edges between glyphs in two clusters in our design to decrease visual clutter.

4. Usability and improvement

All the engineers agreed that the views in our system are easy to understand and use. Six engineers stated that the interactions are nearly perfect, and the system provides necessary interactive functions, such as dragging the glyph, viewing the original images, and entering different hyper parameters. Three engineers praised the convenient compound interactions, such as merging two clusters, which facilitates their exploration. We also obtained valuable suggestions from engineers. Two engineers suggested encoding additional information in the glyph, such as the intensity, color, and position of the light source, to add extra information channels for the lighting scheme design. E_A suggested a new idea about the weighted Jaccard index; that is, combining the values of the important parameters may effectively group images with similar lighting conditions into the same cluster.

8 Discussion and future work

Our work is a multidisciplinary effort, and our approach has received significant feedback from front-line engineers. We innovatively measure the relationships among images by calculating the weighted Jaccard index using parameters, which provides a new angle for engineers to design an appropriate lighting scheme and optimize parameters. For the scalability and effectiveness of our approach, we discuss the problem in two parts: sampling and visualization. The data sampling time and data processing time increase with more sampled images and parameters, but engineers can handle the time-consuming process offline with our graphical software. Our visualization system is tolerant of data expansion; for example, the cluster view encodes an image as a glyph and can simplify a glyph to a point while keeping the cluster structure. The pixel-based images and heatmap of the intersection view are scalable due to the high space stylization. Our system supports the exploration of parameter importance and parameter correlation to meet extra demands from engineers. However, we ignore lighting scheme configuration information and the intermediate images in the algorithm. If we can provide configuration information, such as color and location of the light source, engineers can more successfully identify or choose appropriate lighting schemes. Intermediate images are of high resolution and details of the images are unobservable due to the limited space. Another limitation is that our approach depends on lighting schemes predesigned by the engineers, which requires engineers to have some experience.

In the future, we intend to extend our approach to other more challenging vision tasks, such as defect detection, where the output is an image rather than a numerical value. We will also combine more lighting scheme information in our system to increase its usability, based on the presented feedback.

9 Conclusions

To facilitate the configuration optimization (lighting scheme design and parameter optimization) of MVM, we propose a target-independent approach, visual interactive image clustering, which addresses the two problems simultaneously. We propose a weighted Jaccard index to measure the relationship among images through algorithmic parameters. Based on the explainable measurement, our approach interactively groups images into different clusters with our visualization system (VM-Explorer), and suggests several appropriate lighting schemes with robust parameters. Our approach also supports the exploration of parameter importance and parameter correlation to meet extra demands derived from front-line engineers during close collaboration. We conduct a case study and collect feedback from engineers to demonstrate the effectiveness of our approach.

Contributors

Lvhan PAN designed the research. Lvhan PAN and Baofeng CHANG processed the data. Lvhan PAN, Wang XIA, and Jingwei TANG implemented the system. Lvhan PAN drafted the paper. Guodao SUN helped organize the paper. Lvhan PAN, Qi JANG, Baofeng CHANG, and Guodao SUN revised the paper. Lvhan PAN, Guodao SUN, and Ronghua LIANG finalized the paper.

Compliance with ethics guidelines

Lvhan PAN, Guodao SUN, Baofeng CHANG, Wang XIA, Qi JIANG, Jingwei TANG, and Ronghua LIANG declare that they have no conflict of interest.

Data availability

The data that support the findings of this study are available from the corresponding author upon reasonable request.

References

Afzal S, Maciejewski R, Ebert DS, 2011. Visual analytics decision support environment for epidemic modeling and response evaluation. IEEE Conf on Visual Analytics Science and Technology, p.191-200. https://doi.org/10.1109/VAST.2011.6102457

Albers A, Gladysz B, Pinner T, et al., 2016. Procedure for defining the system of objectives in the initial phase of an Industry 4.0 project focusing on intelligent quality control systems. *Proc CIRP*, 52:262-267. https://doi.org/10.1016/j.procir.2016.07.067

Alonso V, Dacal-Nieto A, Barreto L, et al., 2019. Industry 4.0 implications in machine vision metrology: an overview. *Proc Manuf*, 41:359-366. https://doi.org/10.1016/j.promfg.2019.09.020

Alsallakh B, Ren L, 2017. PowerSet: a comprehensive visualization of set intersections. *IEEE Trans Visual Comput Graph*, 23(1):361-370. https://doi.org/10.1109/TVCG.2016.2598496

Alsallakh B, Micallef L, Aigner W, et al., 2016. The state-of-the-art of set visualization. *Comput Graph Forum*, 35(1):234-260. https://doi.org/10.1111/cgf.12722

Baron ME, 1969. A note on the historical development of logic diagrams: Leibniz, Euler and Venn. *Math Gaz*, 53(384):113-125. https://doi.org/10.2307/3614533

Bergner S, Sedlmair M, Moller T, et al., 2013. ParaGlide: interactive parameter space partitioning for computer simulations. *IEEE Trans Visual Comput Graph*, 19(9): 1499-1512. https://doi.org/10.1109/TVCG.2013.61

Bruckner S, Möller T, 2010. Result-driven exploration of simulation parameter spaces for visual effects design. *IEEE Trans Visual Comput Graph*, 16(6):1468-1476. https://doi.org/10.1109/TVCG.2010.190

Coffey D, Lin CL, Erdman AG, et al., 2013. Design by dragging: an interface for creative forward and inverse design with simulation ensembles. *IEEE Trans Visual Comput Graph*, 19(12):2783-2791. https://doi.org/10.1109/TVCG.2013.147

Comer ML, Delp EJIII, 1999. Morphological operations for color image processing. *J Electron Imag*, 8(3):279-289. https://doi.org/10.1117/1.482677

Eades P, 1984. A heuristic for graph drawing. *Congre Numer*, 42:149-160.

Gleicher M, 2018. Considerations for visualizing comparison. *IEEE Trans Visual Comput Graph*, 24(1):413-423. https://doi.org/10.1109/TVCG.2017.2744199

Godina R, Matias JCO, 2019. Quality control in the context of Industry 4.0. Int Joint Conf on Industrial Engineering and Operations Management, p.177-187. https://doi.org/10.1007/978-3-030-14973-4_17

Golnabi H, Asadpour A, 2007. Design and application of industrial machine vision systems. *Robot Comput-Integr Manuf*, 23(6):630-637. https://doi.org/10.1016/j.rcim.2007.02.005

Kopparapu SK, 2006. Lighting design for machine vision application. *Image Vis Comput*, 24(7):720-726. https://doi.org/10.1016/j.imavis.2005.12.016

Lex A, Gehlenborg N, Strobelt H, et al., 2014. UpSet: visualization of intersecting sets. *IEEE Trans Visual Comput Graph*, 20(12):1983-1992. https://doi.org/10.1109/TVCG.2014.2346248

Luk F, Huynh V, North W, 1989. Measurement of surface roughness by a machine vision system. *J Phys E Sci Instrum*, 22(12):977-980. https://doi.org/10.1088/0022-3735/22/12/001

Martin D, 2007. A Practical Guide to Machine Vision Lighting. Advanced Illumination, Vermont, USA, p.1-21.

Munzner T, 2014. Visualization Analysis and Design. CRC Press, New York, USA.

Ngo NV, Hsu QC, Hsiao WL, et al., 2017. Development of a simple three-dimensional machine-vision measurement system for in-process mechanical parts. *Adv Mech Eng*, 9(10):1-11. https://doi.org/10.1177/1687814017717183

Pretorius AJ, Bray MA, Carpenter AE, et al., 2011. Visualization of parameter space for image analysis. *IEEE Trans Visual Comput Graph*, 17(12):2402-2411. https://doi.org/10.1109/TVCG.2011.253

Pretorius AJ, Zhou Y, Ruddle R, 2015. Visual parameter optimisation for biomedical image processing. *BMC Bioinform*, 16(S11):S9. https://doi.org/10.1186/1471-2105-16-S11-S9

Sadana R, Major T, Dove A, et al., 2014. OnSet: a visualization technique for large-scale binary set data. *IEEE Trans Visual Comput Graph*, 20(12):1993-2002. https://doi.org/10.1109/TVCG.2014.2346249

Sahoo PK, Soltani S, Wong AKC, 1988. A survey of thresholding techniques. *Comput Vis Graph Image Process*, 41(2):233-260. https://doi.org/10.1016/0734-189X(88)90022-9

Sedlmair M, Heinzl C, Bruckner S, et al., 2014. Visual parameter space analysis: a conceptual framework. *IEEE Trans Visual Comput Graph*, 20(12):2161-2170. https://doi.org/10.1109/TVCG.2014.2346321

Torsney-Weir T, Saad A, Moller T, et al., 2011. Tuner: principled parameter finding for image segmentation algorithms using visual response surface exploration. *IEEE Trans Visual Comput Graph*, 17(12):1892-1901. https://doi.org/10.1109/TVCG.2011.248

Wang WC, Guan FN, Ma SY, et al., 2015. Measurement system of gear parameters based on machine vision. *Meas Contr*, 48(8):242-248. https://doi.org/10.1177/0020294015595997

Weng L, Preneel B, 2011. A secure perceptual hash algorithm for image content authentication. IFIP Int Conf on Communications and Multimedia Security, p.108-121. https://doi.org/10.1007/978-3-642-24712-5_9

Wu KL, Yang MS, 2007. Mean shift-based clustering. *Patt Recogn*, 40(11):3035-3052. https://doi.org/10.1016/j.patcog.2007.02.006

Xia JZ, Ye FJ, Chen W, et al., 2018. LDSScanner: exploratory analysis of low-dimensional structures in high-dimensional datasets. *IEEE Trans Visual Comput Graph*, 24(1):236-245. https://doi.org/10.1109/TVCG.2017.2744098

Xia JZ, Chen TX, Zhang L, et al., 2020. SMAP: a joint dimensionality reduction scheme for secure multi-party visualization. IEEE Conf on Visual Analytics Science

and Technology, p.107-118.
https://doi.org/10.1109/VAST50239.2020.00015

Xia JZ, Zhang YC, Song J, et al., 2022. Revisiting dimensionality reduction techniques for visual cluster analysis: an empirical study. *IEEE Trans Visual Comput Graph*, 28(1):529-539.
https://doi.org/10.1109/TVCG.2021.3114694

Xia JZ, Huang LQ, Lin WX, et al., 2023. Interactive visual cluster analysis by contrastive dimensionality reduction. *IEEE Trans Visual Comput Graph*, 29(1):734-744.
https://doi.org/10.1109/TVCG.2022.3209423

Yalcin MA, Elmqvist N, Bederson BB, 2016. AggreSet: rich and scalable set exploration using visualizations of element aggregations. *IEEE Trans Visual Comput Graph*, 22(1):688-697.
https://doi.org/10.1109/TVCG.2015.2467051

Yuan J, Chen CJ, Yang WK, et al., 2021. A survey of visual analytics techniques for machine learning. *Comput Visual Med*, 7(1):3-36.
https://doi.org/10.1007/s41095-020-0191-7

Yuan QM, Zhang H, 2022. Research on the characteristics of light sources in machine vision. *Acad J Sci Technol*, 3(1):1-6. https://doi.org/10.54097/ajst.v3i1.1655

Zhu ZH, Shen Y, Zhu SJ, et al., 2022. Towards better pattern enhancement in temporal evolving set visualization. *J Vis*, early access.
https://doi.org/10.1007/s12650-022-00896-x

Zorcolo A, Escobar-Palafox G, Gault R, et al., 2011. Study of lighting solutions in machine vision applications for automated assembly operations. *IOP Conf Ser Mater Sci Eng*, 26(1):012019.
https://doi.org/0.1088/1757-899X/26/1/012019

List of supplementary materials

1 Mathematical concepts in data sampling
2 Relationships among all concepts in data processing
3 Edge filtering