UNIVERSITÀ DEGLI STUDI DI CATANIA
Dipartimento di Matematica e Informatica
Dottorato Di Ricerca Internazionale In Matematica E Informatica
XXXIII Ciclo

Luigi Claudio Viagrande

Big Data Analysis for Fault Detection in Microprocessors Production

Tesi Di Dottorato Di Ricerca Internazionale

Tutor: Prof. Giovanni Gallo
STMicroelectronics Tutor: Daniele Vinciguerra

Anno Accademico 2019 - 2020

Abstract

This dissertation collects all the research work done by the PhD candidate at the Catania site of STMicroelectronics. The research has been aimed at solving some practical issues coming from the production line in the STMicroelectronics factory. The problem to be attacked is about the classification of Electrical Wafer Sorting (EWS) images. This kind of images are generated daily during the wafer testing phase of semiconductor device fabrication. Classification of these images is relevant to signal defective wafers to be stopped from further processing in the production line. Yield detractors are identified by specific and characteristic anomalies signatures in the images. Unfortunately, new anomalies signatures may appear among the huge amount of EWS maps generated per day. This makes unfeasible, in a real case scenario, to adopt a stable reference set of possible signatures. It is important for the factory to quickly identify and correct the triggering causes of the defect to maintain a high yield rate from the production flow. A correct and quick classification of the images provide clues to the possible cause of the defects and is hence a valuable tool for an efficient production. One may hope in a fixed association between class of defects recognized in the images and problems with the factory processes. The images produced daily in the production line have hence to be fast labeled into categories that would help the system manager to identify the technical problem in the production. Unfortunately it is not possible to resort to a predefined and fixed set of label for the observed defects in the images. New kinds of defects may rise daily in an unpredictable way and in this case new labels have to be produced. The nature of the problem hence asks for a semi supervised classification approach. The solution demonstrated in this Thesis works in successive phases. At first, an unsupervised learning algorithm creates an initial set of clusters from the available images collected off-line. Then, a supervised algorithm classifies the images as they are daily produced into one of the previously created clusters if there is a suitable one. If none of the existing cluster is suitable for the new image a new cluster is added to the collection. This brings out a new issue: the amount of images belonging to newly created classes is, at first, negligible. To alleviate this unbalanced situation one needs to quickly fill these new classes with synthesized images. To make further classification more robust we propose an algorithm that

exploits the shape of the silicon wafer to quickly synthesize new instances of most of the kind of defects that could be encountered. These new synthetic images are hence added to the training set. During the research it become natural to look for methods to detect causal relations between negative production events (alarms) and low yield. A basket analysis approach has been adopted in search for rare association rules mining. To this aim proper customization of the state of the art algorithms has been developed.

On the side of the main research done during the Ph.D. Training period the candidate has been involved in quite different topics. More precisely interesting results have been obtained in the field of design and tools for digital gaming. These results are collected in the Appendix of the Thesis.

Acknowledgements

First of all, I would like to express my gratitude to my advisor, Prof. Giovanni Gallo. With patience, professionalism, and always with a smile he guided me on this path.

I would also like to express my gratitude to my industrial tutor, Daniele Vinciguerra, for having patiently supported me with perseverance and professionalism within all the activities carried out in STMicroelectronics.

I would like to thank Paola Giuffré and Giuseppe Bruno for helping and guiding me during all the activities done in STMicroelectronics.

I would like to thank also Filippo Milotta, a colleague and friend, for supporting me with his professionalism and competence in some of the research activities carried out in STMicroelectronics.

I would like to thank Prof. Filippo Stanco and Ph.D. Dario Allegra for helping and supporting me within the research and activities in the digital games field.

My sincere thanks also go to Caterina Genua, Gabriele Tochino, and all the other STMicroelectronics colleagues with whom I have shared not only my research and activities but also moments of relaxation and conviviality.

Contents

Dedicated to my beautiful wife who always believed in me, to my parents and brother who always support and help me. To all my friends to always be there for me.

Chapter 1

Introduction

STMicroelectronics, founded in June 1987 as SGS-Thomson Microelectronics, is one of the world's largest semiconductor factory with more than 80 offices and 11 manufacturing sites all over the world. Their products are found in many fields of applications, including but not limited to:

- Automotive;
- Industrial;
- Personal electronics;
- Communications equipment, computers and peripherals;

Semiconductor manufacturing requires complex equipment with lots of possible failure points and, for such a large factory like STMicroelectronics, it is essential to be able to guarantee high reliability of the products placed on the market. As the industrial environment of semiconductor manufacturing is very competitive, the factory has to guarantee a very high yield across the entire line and there must be a continuous learning process to keep in place the yield at high levels. Moreover, data from manufacturing plants arrive daily and, given the huge amount of data we fall within the context of Big Data. Among the many types of data collected by the factory, we find Electrical Wafer Sorting (EWS) maps. These images are generated during the wafer testing phase performed in semiconductor device fabrication. Electrical wafer sort is the last stage of wafer fabrication. A wafer is a round-shaped support containing several "dies" and each die is subject to electrical tests. After testing, good dies are cut out from the wafer and sent to the package phase. Instead, bad ones were literally "inked" to be easily recognized and discarded when dies are extracted from the wafer. Today, defective dies are not inked anymore, as this can be done digitally, employing maps that can be used for masking good and bad portions of the wafer. Maps represent dies on the wafer and, according to a statistical binning approach, they can have several values (i.e., good or failed during test stage 1, stage 2, and so on). This dissertation collects all the research work done by the Ph.D. candidate in STMicroelectronics site in Catania. The factory needed some systems that could optimize production to increase profits while reducing losses. Among these is the need to find a way to quickly group the EWS maps that arrive daily from production. This task was previously carried out by technicians and it was tedious and time-consuming. Moreover, the classification may not be accurate as it was subject to the technician's personal vision. The addressed problem can be represented as a semi-supervised case, since the analysed data do not contain any labels and we don't know how many classes there will be. This is hence an open set problem where new classes could potentially arise daily. It was therefore decided to split the problem into two parts.

- Unsupervised clustering: Starting from images collected in few months we created an initial knowledge base of homogeneous classes using an unsupervised clustering method.

- Supervised classification: After the knowledge base definition, we were able to take advantage of some supervised classification methods that would have compared the new images with the knowledge base previously acquired.

A secondary problem showed up as a consequence of the issue just described. As mentioned, images are generated daily and new classes may generate from new images. However, the new classes may contain only a few images at first, insufficient to provide accurate automatic classification. It was therefore necessary to analyse data augmentation methods already known in literature to develop an augmentation method suitable for the needs of STMicroelectronics and which would be well suited to the type of data analysed.

Another problem faced during the Ph.D. years is the association rules learning. In addition to cataloging the defects present in the wafers, the factory is very interested in researching the triggering causes. It is common to encounter technical or human problems during production, called alarms. The analysis of these alarms and the causes triggering a negative yield of the products can generate rules that uniquely associate one or more alarms and a non-optimal performance of their products. With this dissertation we want to contribute to the growth of STMicroelectronics, demonstrating the enormous interest shown by the factory towards the work done, the benefits obtained, and presenting some possible use cases and how our research evolved, following the actual use of the proposed solutions in production.

1.1 Dissertation structure

This dissertation is structured as follows:

In chapter 2 the previous works related to unsupervised learning are presented and the first part of the semi supervised EWS maps classification tool, created to address the problem of image classification, is presented.

In chapter 3 the previous works related to modern supervised classification methods is presented, and supervised methods are also presented which, alongside the clustering method, contribute to the semi supervised EWS maps classification tool.

In chapter 4 both modern and classical augmentation methods are discussed. The development of the semi supervised EWS maps classification tool brought to light the problem of under-sampling of new classes and, thus, the Image sectors replacement tool is then exposed as a solution.

In chapter 5 the principles of basket analysis and association rules learning are explained. STMicroelectronics, in addition to the classification of defects in the EWS images, needed to investigate the triggering causes for these defects. The most famous algorithms that deal with similar problems will be treated and Low yield Apriori searching tool is presented as the solution.

In the final chapter 6 the conclusions of this dissertation will be drawn and the benefits of our research and future works will be highlighted.

1.2 List Of Publications

In this dissertation, we present a real use-cause, co-authored and published in an international conference.

- L. C. Viagrande., F. L. M. Milotta., P. Giuffrè, G. Bruno., D. Vinciguerra.,and G. Gallo. "Semisupervised Classification of Anomalies Signatures in Electrical Wafer Sorting (EWS) Maps". In: Proceedings of the 15th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications - Volume 5: VISAPP, INSTICC. SciTePress, 2020, pp. 278–285. isbn: 978-989-758-402-2. doi:10.5220/0008914402780285;

- This article shows the semi-supervised infrastructure for classifying EWS images coming from STMicroelectronics factory.

The articles published on secondary activities carried out during the Ph.D. period will be listed below.

- L. C. Viagrande, D. Allegra, and F. Stanco. "Curve Design Studio: Bézier Curve Integrated Tool for Video Game Development". In: Smart Tools and Apps for Graphics - Eurographics Italian Chapter Conference. Ed. by S. Biasotti, R. Pintus, and S. Berretti. The Eurographics Association, 2020. isbn:978-3-03868-124-3. doi:10.2312/stag.20201236;
- L. C. Viagrande, D. Allegra, and F. Stanco. "Point to Culture: a Point to Click Framework for Serious Games in Cultural Heritage". In: Smart Tools and Apps for Graphics - Eurographics Italian Chapter Conference. Ed. by S. Biasotti, R. Pintus, and S. Berretti. The Eurographics Association, 2020. isbn: 978-3-03868-124-3. doi:10.2312/stag.20201248;

- Curve Design Studio is a tool built with Unity3D game engine. It is an open and easily extendable solution to address the problems related to curves that can be identified in the production process of a video game. Point To Culture is a framework built with Unity3d as well. It is an immediate solution to the creation of serious games oriented to cultural heritage as it offers all the necessary tools and the possibility to extend the desired functions.

Chapter 2

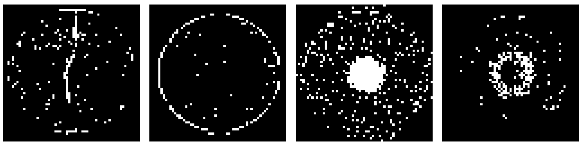
Semi-Supervised EWS Maps Classification Tool: Unsupervised clustering

2.1 Introduction

Since many yield detractors can impact production at the same time, device engineers have to spend a lot of time analysing EWS data to identify every yield detractor and relative affected wafers before proceeding to in-depth analysis and data mining to identify root causes.

Usually, yield detractors can be identified by specific and characteristic patterns, named EWS map signatures (i.e., scratch, ring, spot, or wheel, as shown in Figure 2.1). These patterns are useful for investigating the root causes that could be, for instance, related to an equipment component failure, a drifting process, or an integration of processes [1]. Unfortunately, new anomaly signatures may appear among the huge amount of EWS maps generated per day (i.e., we are facing an open set problem). Hence, it's unfeasible to define just a finite set of possible signatures, as this will not represent a real use-case scenario. The automatic labeling of old and new anomalies represents an interesting research issue with relevant industrial applications.

We are within the scope of unsupervised learning, and we need to apply a clustering strategy. Since the concept of cluster in data analysis has been introduced, it has spanned through a wide range of disciplines [2, 3]. Many real-world problems, indeed, found their solution on cluster analysis. In this work, the "objects" to cluster



(a) (b) (c) (d)

Figure 2.1: EWS maps showing the following characteristic anomalies signatures: (a) Scratch, (b) Ring, (c) Spot, and (d) Wheel. White pixels identify defective dies.

are real EWS maps acquired by STMicroelectronics. We wish to aggregate the EWS maps accordingly to their anomaly signatures. In practical clustering applications, it is not known in advance how many clusters have to be treated, and their number may also change when different days of production are considered.

Although in certain cases the number of clusters is already known, this does not apply in our case, as said before.

Many studies have been carried out to estimate the optimal number of clusters for a given clustering task, but this field is very challenging, and existing methods still have drawbacks [4]. In our application, the number of clusters is daily increased upon dynamic data analysis. In the proposed method, we have looked into the field of aggregative clustering, one of the earliest and most widely used clustering strategies [5, 6]. We have evidence of aggregative clustering since 1950 [7].

Once clustering phase is completed, we will have a dataset of EWS maps where each image will be assigned to a specific and unique cluster. Then, as in a supervised approach, we may leverage the outcome of clustering phase considering the assigned cluster-ID as label for training a classifier. Indeed, this hybrid approach is defined semi supervised learning [8, 9], and it is particularly meaningful as the manual classification could lead to different results among different operators, who are biased by subjective interpretation of the anomalies signatures. Clustering analysis was leveraged for anomalies signatures classification in recent works [10, 11, 12, 13]. However, all these works applied supervised learning assuming a finite set of just 9 anomalies signatures. They employed the WM-811K dataset [14], which is made of 811,457 wafer maps collected real-world fabrication. Images in the WM-811K dataset are similar to EWS maps we employed in this work: in the former case, images are labeled, while in the latter they are not. One of the most recent related works on anomalies signatures retrieval was focused onto wafer defect maps (WDM) classification [15]. Even if WDM images look similar to EWS maps, we remark that defectivity analysis is a test phase performed some steps before the EWS one. Differently from the present work, a labeled dataset with a set of finite classes was employed for WDM classification, too.

In sum the contributions of this first work are:

- **Semi supervised classifier:** a new semi supervised approach for classifying anomalies signatures in EWS maps is presented, by combining an unsupervised approach using a hierarchical clustering algorithm to create the starting Knowledge base, and a supervised one, presented in chapter 3, through a classifier trained leveraging clustering phase (Figure 2.2);
- **Daily update:** our dataset can be daily increased, and the classifier is dynamically updated considering possible new created clusters. The workflow of our solution can be resumed in: daily arrival of EWS maps, clustering of newcomer images into previously created clusters, possible creation of new clusters, anomalies signatures classification;
- **Variable number of anomalies signatures:** we are not considering a fixed number of anomalies signatures, and the leveraged dataset does not contain any label. This represents the typical scenario of real use-case industrial applications.

The goal of this work was to create a tool to make as automatic as possible the recognition of wafer anomalies signatures. This is meaningful as upon classification the industrial system can be able to automatically choose (or at least suggest) either to discard a wafer or to ship it to the customer. The proposed method can also grant benefits like reduction of wafer test results review time, or improvement of processes, yield, quality, and reliability of production using the information obtained during clustering process. Image dataset used in this work is made up of EWS data gathered for months from STMicroelectronics and that has been converted into binary images. For simplicity, we assume to handle binary EWS maps, where white pixels identify failed dies, while black pixels the good ones. White pixels represent bad portions in the wafer, as can be seen in Figure 2.1. In the present case, they represent defective dies who have not passed at least one of the EWS tests. Black pixels represent

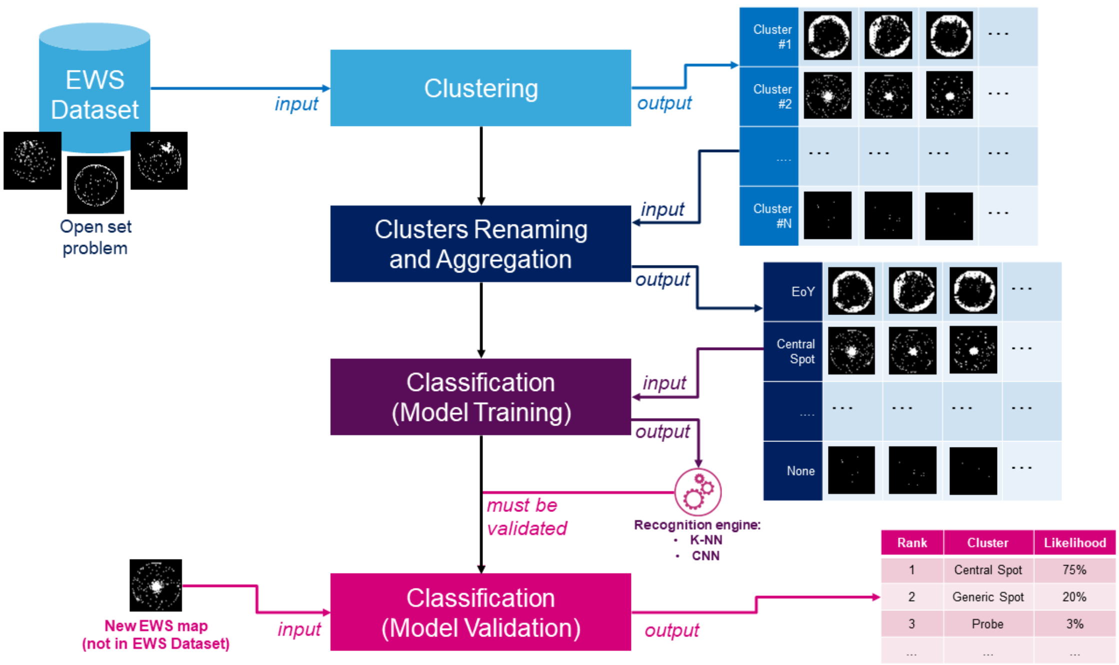


Figure 2.2: Overview workflow of the proposed approach.

good dies who have passed all the EWS tests. The images are therefore scaled to a resolution of 61×61 without losing essential information. The clustering engine developed is based on agglomerative hierarchical clustering. The proposed solution is proven to be the most effective for the problem we had to face and was chosen after a careful analysis of other algorithms known in the literature. The algorithm takes as input a group of descriptors that can effectively represent the images without losing information since it would not be feasible to use raw images as input for the algorithm. For the present case, we extracted *principal components* among others as they have proven to be more effective than other types of descriptors for the speed of extraction and use as well as for the high accuracy shown in the clustering phase. The present chapter is structured as follows: in 2.2 the history of cluster analysis will be discussed together with its growth over time and the algorithms covered in this dissertation will be presented. Finally, the use of cluster analysis in the field of microelectronics will be discussed. In 2.3 the structure of the clustering algorithm used for STMicroelectronics purposes will be presented. In 2.4 the results of the experiments conducted on multiple algorithms and descriptors will be presented. Finally, in 2.5 the contribution highlighted throughout the chapter will be briefly exposed.

2.2 Background and Related Works

Clustering is the set of data analysis techniques that allows the grouping of homogeneous elements. Cluster analysis has always been widely exploited by the scientific community. From 1960, we can see a significant increase in the number of papers dealing with the topic [16]. From that time, this field has grown and evolved and it has spanned through a wide range of disciplines [2, 3, 17]. Preliminary studies with cluster analysis have been conducted in the biological and zoological fields and it was used to group animals and plants. Given the effectiveness of the method, cluster analysis has also been successfully used later in the field of medicine for grouping disease and illness, until further expanding to other fields such as sociology and criminology, anthropology and archaeology, geology, geography, and remote sensing. In the early eighties, cluster analysis has expanded to the IT field and it was used mainly for grouping images, documents and market analysis [17]. There are three elements that characterize the cluster analysis and are therefore necessary to perform it [18]:

- **a proximity measure** to analyse the similarities between patterns;
- **a clustering algorithm** to perform the analysis;
- **a quality measure** to evaluate the outcome of the analysis;

2.2.1 Related works

Clustering algorithms could be divided into traditional and modern ones [19]. Among traditional models, we find algorithms based on partitions. K-means, proposed by MacQueen et al. in 1967 [20], is one of the most popular algorithms based on partitions. The goal of K-means is to partition the data into a number of clusters, k. Algorithms based on partitions treat the center of data as the center of the specific cluster. K-means updates the center of clusters iteratively until a convergence criterion is reached. K-means is very quick and efficient in general, but it is extremely sensitive to outliers. Another traditional model is given by algorithms based on hierarchy. The idea behind this model is to consider each data point as a separate cluster and to construct the final clusters by merging each cluster with its closest neighbour. Another widely exploited approach is the reverse, that is, start with a single cluster containing all the data points and split it. Hierarchical clustering algorithms [21] belongs to this model. These algorithms are generally very scalable, but they are not as quick as partitions based algorithms. Furthermore, clustering methods are still used and we have evidence about it [7, 5, 22, 23, 24] but research into this field began long before [25, 26, 27, 28, 29] and it never stopped, with the scientific community investigating this topic to the present day [30, 11, 13]. Not only. Cluster analysis was lately used effectively in the microelectronic industry [31, 10] hence demonstrating the effectiveness and the need to direct preliminary studies on the clustering of non-labeled EWS maps toward an unsupervised learning approach making it hence a starting point to later direct the research towards a supervised approach.

As previously mentioned, three elements characterize the cluster analysis [32]:

- **a proximity measure** to be able to estimate a distance between the data. The most common proximity measure is the Euclidean measure;
- **a clustering algorithm** to perform the analysis. In the present dissertation, an agglomerative hierarchical clustering algorithm is presented. As a first step, a dissimilarity matrix is built starting from the proximity measure. Each data is hence represented as a point at the bottom of the dendrogram. The closest set of points is merged or agglomerated at each step of the algorithm until a cluster with all the data is obtained. At each step of the algorithm, the dissimilarity matrix is updated [33]. An example of dissimilarity matrix is shown in 2.1 To perform the merging, a linkage method is used. The oldest method is the single linkage and it defines the minimum distance between two clusters as the shortest distance between a member of one and a member of the other as it is defined by the equation (Eq. 2.1);

$$\min\{d(a,b) : a \in A, b \in B\} \tag{2.1}$$

- **a quality measure** to evaluate the outcome of the analysis. There are several criteria to perform this evaluation and they are mainly classified as external or

Table 2.1: Example of dissimilarity matrix, showing the distance between each pair of points

	A	**B**	**C**	**D**	**E**
A	0	0.2	0.6	0.5	1
B	0.2	0	0.3	0.4	0.8
C	0.6	0.3	0	1	0.7
D	0.5	0.4	1	0	0.4
E	1	0.8	0.7	0.4	0

internal whether the proposed validation set comes from an external training dataset or the same database used for the clustering. Another measure is called relative and it compares the obtained clusters with a series of predefined clustering schemes;

Automated analysis of defect wafer in microelectronics industries has been investigated for many years to monitor the production activity and take immediate actions on the root causes of failure production points to increase performance and yield. Hence, with the growth of factories and an increasing amount of data over time, the use of automatic analysis methods has become necessary [34].

Spatial Signature Analysis (SSA) [27] is one of the early procedures made up for wafermaps defect clustering and signature classification. In the past, defect analysis was made by engineers that had to analyse wafer surfaces with optical instruments. Given the spatial representation of the defects on the surface of the wafer, they associated a production problem with each specific defect. The SSA procedure begins with creating groups of pixels (clusters) based on several factors. These clusters are assigned to a specific set depending on morphology and proximity with other neighboring clusters. Since each set has unique characteristics, each cluster of pixels contains unique descriptive features in relation to the belonging set.

These features are hence used by a classifier to perform the classification. The SSA procedure takes advantage of a pair-wise fuzzy KNN algorithm. In practical terms, SSA was able to quickly turn wafermaps optical defect data into useful information and to provide high performance when classifying defects.

More modern approaches have been developed over the past decade. In [10] a methodology based on unsupervised learning for clustering of wafer spatial signatures is proposed. Their solution performs in three steps. First, they extract descriptors from wafer spatial signatures with sparse regression. As a second step, a hierarchical clustering algorithm is performed for grouping similar wafers. As the last step, a modified L-method is performed over the hierarchical clustering result to determine the right number of clusters. In [31] a method is proposed that uses unsupervised learning algorithms like hierarchical clustering for making clusters from spatial signatures of failures, taken with the analysis of the binary testing results of each die in the wafers and finally identifying systematic failure patterns.

2.3 Hierarchical clustering algorithm structure for electrical wafer sorting (EWS) maps

The clustering algorithm chosen to face STMicroelectronics issue is an agglomerative hierarchical one. Many other algorithms have been tested but without obtaining good results. Among them, we find K-Means and divisive hierarchical clustering. The steps that led to the creation of the clustering infrastructure is shown below in the main key points which will subsequently be explained in detail:

- **Descriptors extraction** to search the best-known descriptors to represents unlabeled data;
- **Preliminary clustering phase** to test the various algorithms with a small sample of data;
- **Knowledge base definition** to establish an initial knowledge of known signatures;

2.3.1 Descriptors extraction

Although many descriptors have already been successfully tested on supervised problems [11], given the unlabeled characteristic of STMicroelectronics dataset, we decided to investigate two more general effective descriptors: Local Binary Pattern (LBP) and Principal Components(PC).

Local Binary Pattern (LBP): The Local binary pattern is defined as a grey scale invariant texture operator and it is widely used in image analysis. Given a grayscale image, the operator compares the 3×3 neighborhood of each pixel with this central pixel value and transforms the result to a binary number. A representation of the image is hence obtained through a histogram of these binary numbers [35].

Principal Components (PCs): The PCs are computed through the Principal Component Analysis (PCA). The PCA is an orthogonal linear transformation that modifies the data to a new coordinate system highlighting their similarities and differences [36]. Images with a resolution of 61×61 pixels are "vectorized", that is they are reshaped into a vector of 3,721 pixels [37]. We decided to keep as many PCs as needed to have at least the 80% of variance retain. Typically, only 15 to 20 PCs survive over the 3,721 original ones. These PCs are used as a descriptor for the clustering phase.

2.3.2 Preliminary clustering phase

During the preliminary clustering phase, we collected the first dataset of 296 images, with a resolution of 61×61 pixels. This dataset was the only one to be manually labeled by a team of experts of STMicroelectronics. This was necessary to validate the preliminary clustering outcomes. As introduced in 2.1 we analysed some clustering techniques, including K-means and divisive hierarchical clustering but we focused on the aggregative hierarchical clustering algorithm. Indeed, the main purpose of this phase was to select the clustering algorithm, and we chose the aggregative hierarchical clustering for its results obtained on clustering the 296 images on 5 clusters. Hierarchical clustering is an unsupervised algorithm that subdivides the dataset into partitions leveraging a distance function. It produces a hierarchical representation, with the lowest level of the hierarchy counting n clusters, where n is the total number of observations. Instead, at the top level, we have a single cluster containing all the observations. Hierarchical clustering comes with many linkage methods to perform the clustering [38]. Linkage methods task is to define how the distance between two clusters is measured and it is hence important as it also defines how to assign an observation to one of the many available clusters. The Ward linkage [39] is the method used in this work. It is used to minimize the variance of the clusters being merged. Using the Ward linkage an error function is defined for each cluster as the average distance of each observation in a cluster to the centroid of the cluster. The distance between two clusters is defined as the error function of the unified cluster minus the error functions of the single clusters. The only hyper-parameter we need to set in the algorithm is the number of clusters. Since it is an open set problem, we had to find a good compromise between how many clusters to generate and how coherent they should be. There are many techniques for determining the number of clusters [40], including the calculation of the within-cluster sum of squared error (WCSSE) [41], used in our work and it is defined by the following equation:

$$WCSSE = \sum_{k=1}^{K} \sum_{i \in S_k} \sum_{j=1}^{P} (x_{ij} - \bar{x}_{kj})^2 \tag{2.2}$$

where S_k is the set of observations in the $k-th$ cluster, and $\bar{x}_{kj}$ is the $j-th$ variable of the cluster for the $k-th$ cluster found with the clustering algorithm. In our experiments, the value of this parameter was empirically set to 6,000. We visually confirmed that, with this chosen value, we have a good compromise between clusters generated and intra-cluster variance.

2.3.3 Knowledge base definition

In a typical scenario of industrial use-case, the dataset is daily increased with new EWS maps. Hence, in a process of knowledge base definition, we gradually increased the size of our dataset until it counted 10,000 images, with a resolution of 61×61 pixels. It is called "knowledge base" as it represents our core knowledge about the possible anomalies signatures (i.e., the number of clusters) known until each daily update. We therefore dynamically proceeded to test our clustering procedure on the incrementing dataset. Eventually, we obtained 10 clusters. Then, clustering outcomes were once again validated by internal experts. The dataset of 10,000 unlabeled images has been used as a starting knowledge base for the classification phase.

2.4 Experimental results

After deciding which descriptors to use (i.e., Local Binary Pattern - LBP, and Principal Components - PCs) we proved the goodness of the descriptors combining them with the use of each designated algorithm (i.e., K-Means, divisive and aggregative hierarchical cluster). Outcomes are reported in Table 2.2. We observed that the quality of hierarchical clustering, when combined with PCs, outperformed the quality of K-means. We also found that aggregative hierarchical clustering performs better than a divisive one.

Table 2.2: Clustering outcomes when changing clustering algorithm and EWS maps descriptor (LBP: Local Binary Pattern, PCs: Principal Components). Precision is computed as $TP/(TP + FP)$. The best result is highlighted in bold.

Clustering	**Descriptor**	**Precision %**
KMeans	**LBP**	48.98
KMeans	**PCs**	70.6
Hierarchical clustering divisive	**LBP**	58.44
Hierarchical clustering divisive	**PCs**	79.05
Hierarchical clustering aggregative	**PCs**	**90.20**

A visual representation of the goodness of the clustering methods used is given comparing Figures 2.3, obtained through K-Means with PCs descriptors and 2.4 obtained with divisive hierarchical clustering algorithm combined with LBP descriptors. As shown, PCs outperform LBP even if we employ K-Means instead of hierarchical clustering. The LBP operator performs better on images having a well-defined pattern but fails to find other kinds of defects. Moreover, the clustering through PCs performs very quickly, as the process runs in no more than five minutes per day, while the LBP operator significantly increases the processing time to few hours a day, which makes it not feasible for STMicroelectronics needs.

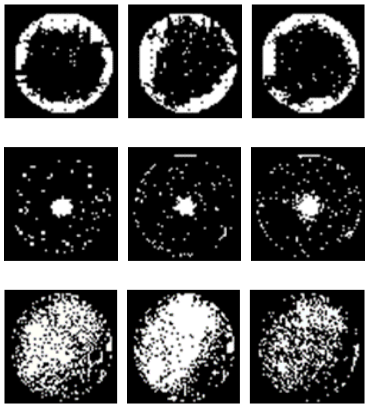


Figure 2.3: Three clusters (one per row) obtained using K-Means with Principal Components (PCs). This is an example of good clustering.

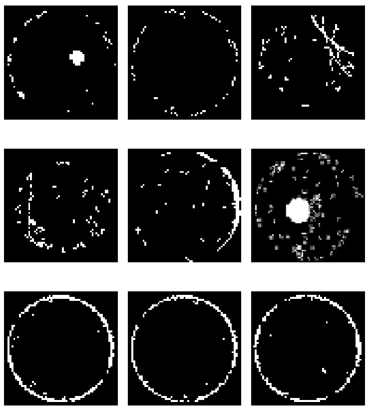


Figure 2.4: Three clusters (one per row) obtained through divisive hierarchical clustering with Local Binary Pattern (LBP). This is an example of bad clustering: first and second clusters contain different kinds of anomaly signatures. Only the third cluster looks fine.

2.5 Conclusions

In this chapter, we presented several unsupervised clustering methods. We defined the Hierarchical clustering algorithm as it is used by STMicroelectronics. We show how the agglomerative Hierarchical clustering algorithm combined with PC descriptors outperforms other methods and descriptors. The algorithm has been used to make a first knowledge base of 10 classes. Then, with each daily update, the same algorithm is used to cluster new images that do not fall into any other clusters and which are therefore discarded during the classification phase, thus increasing the knowledge base with new clusters. The supervised classification phase will be discussed in chapter 3.

Chapter 3

Semi-Supervised EWS Maps Classification Tool: Supervised Classification

3.1 Introduction

After the clustering phase and the creation of a knowledge base, the use of a supervised method was necessary for the classification of new daily images. In the beginning, the images in the factory dataset did not allow the use of more accurate methods such as neural networks. The dataset had little more than ten thousand images and was enlarged by, approximately, one thousand images per day. It was therefore decided to use K-nn classification algorithm, providing a less modern approach but with a good percentage of success. Later on, with the growth of the dataset, it was possible to use a more modern approach such as Convolutional Neural Networks (CNN) with a ResNet-18 architecture [42]. For both cases, PCAs with the same configurations as for the clustering phase were used. In addition, the created algorithm allows discarding images that have failed to classify correctly among previously created clusters and which will therefore be clustered again thus contributing to the growth of the knowledge base with new upcoming clusters. The present chapter is structured as follows: in 3.2 the history and background of supervised classification will be discussed. In 3.3 the structure and configuration of the implemented k-nn algorithm will be presented. In 3.4 the configuration of the neural network used in the present work will be exposed. In 3.5 the results of both classification methods used will be treated. In 3.6 a small summary of the chapter and the work done is presented.

3.2 Background and Related works

Supervised learning usually refers to a classification task where there is the need to build a classifier that can correctly predict the class of an object given the observations of other objects whose classes are already known. We have evidence of supervised learning since 1992 when it has become a standard solution in many application fields [43].

3.2.1 Related works

Since 1992, many supervised learning methods have appeared and slowly spread, including K-nn. K-nn algorithm assigns to incoming test samples the label of the nearest sample in the training set and is therefore considered the simplest nonparametric classification algorithm [44, 45]. Neural computing is proposed as a valid solution for pattern recognition problems and it has begun to spread widely for its simplicity of construction and for being inspired by studies on biological networks bringing this technique closer to human reasoning [46, 47].

Nearest neighbor is a technique already investigated since the late 1960s [48]. The purpose of the k-nn algorithm is to assign a label to a new observation x based on the majority of the labels of the nearest k observations. In other words, with $k = 1$, given n correctly classified points, the point x'_n with

$$x'_n \in (x_1, x_2, x_3, ..., x_n) \tag{3.1}$$

is the nearest neighbour of x if:

$$min\, d(x_i, x) = d(x'_n, x)\; i = 1, 2, 3, ..., n \tag{3.2}$$

With large number of samples, it is better to use $k > 1$ ranking, hence, on the majority of votes. With k-nn there is the need to look for the optimal number k. Using k too large or too small will probably cause wrong classification, the first one because it will take into account samples far away from the analysed point while the second one will be affected by outliers [45]. K-nn is a highly exploited algorithm and it is still studied and used today despite more accurate and powerful classification methods like neural networks [49, 50, 51] Neural networks began to spread in the early 1990s as a solution to classification and pattern recognition problems. Its simplicity of construction and conceptual proximity to biological networks have allowed its widespread diffusion over time and to a multitude of fields of application: medicine, biology, signal processing to name a few [52, 53]. Convolutional neural network (CNN) is one of the first types of Neural Network that has become very popular for its effectiveness in classifying images. In the beginning, a Convolutional Network was a multi-layer network, and all the connections were adaptive and trained with back-propagation. Pre-processing was minimal [54, 55, 56]. Over the years, research on this technology has continued and convolutional neural networks are now used in many fields and practical computer vision applications such as object recognition, semantic segmentation, object detection, video analysis, medicine, and, of course, micro-electronics industry [57, 58, 59].

There is much evidence of effective use of supervised learning techniques and models in microelectronics industry for classification of wafer maps using both neural networks and other supervised learning approaches. In [60] a modified k-nn algorithm is used to classify different kind of scratches in wafer maps. This approach takes advance of feature extraction such as distance, slope, and curvature, used as input for their novel k-nn algorithm. In literature, we also have evidence of mixed approaches. In [61] a Convolutional autoencoder is used for feature extraction over two-dimensional grayscale images to feed a k-nn algorithm for the classification phase. In [62] a Convolutional network is presented for wafer map defect pattern classification and image retrieval. Moreover, it demonstrates how it is possible to correctly classify real wafer maps using only synthetic images for training. In [12] a mix of classifiers is used:

- **Logistic regression**;
- **Random forest**;
- **Gradient boosting machine**;
- **Artificial neural network**;

These classifiers were trained using a mix of extracted features:

- **Density**;
- **Geometry**;
- **Radon-based**;

Then a soft voting ensemble technique assigns certain weights to the classifiers according to their prediction accuracy. In addition, classification through supervised learning has already been sought by another research team from STMicroelectronics. They used a submanifold sparse convolutional network in combination with an ad-hoc augmentation procedure for imbalanced classes [63]. In general, supervised learning proves to be of incredible scientific and practical relevance for microprocessor manufacturing factories and, over time, it has proven to be a quick and effective method to address wafer map classification problems.

3.3 K-nn algorithm structure for Electrical Wafer Sorting (EWS) maps classification

As mentioned in 3.1 a routine using k-nn algorithm has been developed that dealt with classifying the new images that had arrived daily and that, at the same time, dealt with discarding the images that did not clearly belong to one of the classes present in the knowledge base. The images discarded by the k-nn algorithm are hence fed to the hierarchical clustering algorithm described in 2.3 and are therefore divided into clusters that will increase the knowledge base. The decision to use the k-nn as an initial classification algorithm was made due to the need to use a procedure that could quickly adequately classify the new daily images and increase the knowledge base without the long waiting times given by the training phase necessary for the neural networks.

3.3.1 K-nn settings

The procedure that exploits the use of a hierarchical clustering algorithm and which is described in 2.3 allowed to create an initial knowledge base of 10 classes over

10,000 images. The k-nn algorithm used for classification decides whether a new image may fall into the already existing knowledge base or whether it should possibly be incorporated into a new cluster which will increase the knowledge base. The k-nn procedure deals with comparing a number k of samples closest to the query observation. The label of the new sample depends on the majority of the closest samples [64]. In our solution, the distance is evaluated with the Euclidean distance between the descriptors of the samples described in 2.3.1. The optimal number k of samples was empirically fixed at 15. Given the need to check daily for new defects to increase the Knowledge base it was appropriate to introduce an affinity percentage. If at least 10 over 15 neighbors of the query sample belong to a certain class, that sample will be assigned to that class as well, otherwise, it will be discarded. The discarded images are then passed back to the hierarchical cluster algorithm, eventually forming new classes that will increase the knowledge base.

3.4 CNN classification

Once the knowledge base grew (58,038 images, with a resolution of 61×61 pixels split in 85 classes) it was appropriate to use a more accurate classification technique than the k-nn approach. A Convolutional neural network classifier was trained with a ResNet-18 architecture [42]. It was decided to use this architecture because we have evidence of its effective use for the same purpose [12]. The dataset counted 58,038 images and it was splitted in 46,431 (80%) images for the training set, and 11,607 (20%) images for the validation set. As can be seen, in (Table 3.1) the dataset is not balanced. This aspect can be considered a quality of our dataset because it is common for real use-case industrial applications to face some anomalies more than others. To solve the problem of dataset imbalance it was decided to exploit some data augmentation techniques already known in literature. This process was necessary to generate fake images until all classes had reached the minimum of 100 samples. Starting from real images of under sampled clusters, the augmentation consists of a random combination of one or more of the following techniques:

- **Noising:** some white pixels where randomly put inside the image (Gaussian noise).

Table 3.1: Dataset imbalance in Training and Validation sets. The relative quantities (Rel Qty) of the 4 biggest clusters have been reported. *Cluster 26* is considered by STMicroelectronics experts to be the group of wafers without any anomalies signatures (i.e., good wafers).

Cluster ID	***26***	***34***	***12***	***6***	***Others***
Training Set Rel Qty	32.20%	16.45%	7.13%	4.55%	<4.00%
Validation Set Rel Qty	32.14%	16.53%	7.08%	4.55%	<4.00%

- **Rotation:** the image where rotated randomly of 90, 180 or 270 degrees, clockwise or counter clockwise.
- **Flipping:** the image were horizontally or vertically flipped.

3.5 Experimental results

The results of the classifications performed by the algorithms are reported in (Table 3.2). The descriptors used by the algorithms are the PCs, exactly as for the clustering phase. It can be noted that the k-nn algorithm with the addition of the affinity percentage parameter manages to obtain a fairly high classification precision actually reaching a percentage of 90.55%. As expected, the use of CNN overcomes the performance of k-nn with an accuracy percentage of 95.87%. Since we are dealing with an unbalanced dataset, we also computed for CNN the more robust F1-score, which is equal to 92.18%. Considering we started from a not labeled dataset, our results sound comparable with the ones shown in [12], where they obtained 96.93% of precision and 96.71% of F1-score with only 9 classes [14], instead of the 85 employed in this work. Another good quality of the proposed method is its proved rotation invariance (Figure 3.1).

Table 3.2: Classification outcomes comparing K-Nearest Neighbours (KNN) and Convolutional Neural Network (CNN). EWS maps selected descriptors are the Principal Components (PCs). Precision is computed as $TP/(TP + FP)$. The best result is highlighted in bold.

Classification	**Descriptor**	**Precision %**
KNN	**PCs**	85.33
KNN with affinity percentage	**PCs**	90.55
CNN	**PCs**	**95.87**

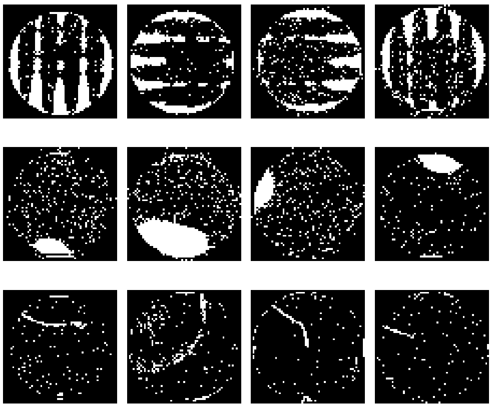


Figure 3.1: Proposed clustering method is proved to be rotation invariant.

3.6 Conclusions

In this chapter, we presented some supervised classification techniques. The typology of classification problem faced in STMicroelectronics made it necessary to use a less effective method such as k-nn compared to the use of a neural network but more suitable for an initial classification in which there were few classes. We presented the configuration of the k-nn algorithm currently in use in STMicroelectronics and demonstrated its classification effectiveness. Later, after increasing the knowledge base in STMicroelectronics and collecting 85 classes it was decided to use a more modern and accurate neural network classifier reaching a higher classification accuracy than the one obtained with k-nn algorithm. As future works, we planned to investigate the performance of other CNN architectures. We are also designing a comparison study with a two-fold purpose: consolidate outcomes shown in this architecture employing the WM-811K dataset, and exploring the existence of any correlation with test phases before the EWS (e.g., relatively to Wafer Defect Maps - WDM).

Chapter 4

Image sectors replacement tool

4.1 Introduction

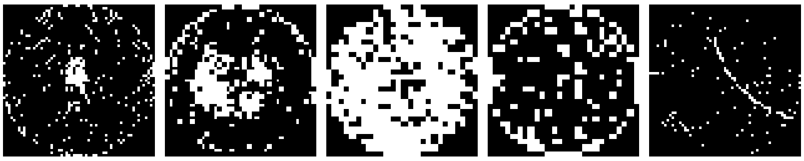
Data augmentation is the procedure of generating fake data samples with the manipulation of real data available in the dataset. It is a label preserving procedure and it is often needed to improve the accuracy of classifiers when the number of samples is not balanced between classes. In this work, we focused on the choice of the right Data Augmentation method for the semi-supervised classification problem we already addressed in chapters 2 and 3 and it is hence part of the work already covered in [65] which focused on a very specific kind of data coming from STMicroelectronics called Electrical Wafer Sorting (EWS) maps. These images are generated during the wafer testing phase performed in semiconductor device fabrication. After testing, damaged dies in a wafer may generate a specific signature that will be represented on a map and need to be correctly classified to allow technicians to analyse the problems that may occur in production. One of the most common problems that may occur during the classification of images is the lack of elements in some classes. This could result in a loss of accuracy when classifying images that should fall into classes with few elements. The extent of the problem can be easily recognized in a factory as big as STMicroelectronics where there are thousands of new daily images and they need to correctly classify them quickly. Moreover, in this industrial context, new signatures may arise daily which is reflected in the creation of new classes with very few samples at first. It is, therefore, possible that other images belonging to the newly created classes are incorrectly classified. There is hence the need to populate these classes as soon as possible with images that are synthesized without introducing any type of bias to raise the classification accuracy for these classes. If the classes are not equally distributed an unbalanced dataset will be created and some majority classes are usually obtained to the detriment of classes with fewer elements. The majority classes, in this case, will tend to orient the classification towards them, altering the final results [66]. Many augmentation methods are already known [67] divided according to the type of manipulation to which the images are subjected. Older Data Augmentation methods, working in the input space, include direct geometric image manipulation like flipping, cropping, rotation, translation, and noise injection. Subsequently, we go into the photometric field with color space transformations or kernel filters to provide special effects to the images like blurring or sharpening. Mixing images is another well-known strategy. However, it is not always applicable as it may produce useless images. Random erasing is another manipulation technique. Removing a part of the image forces the model to find other descriptive characteristics. Recently there has been an evolution of augmentation methods based on deep learning. Instead of applying manipulations to images in the input space, deep learning augmentation acts in the feature space. Neural networks provide a powerful mean for creating fake images starting from high-dimensional inputs and mapping them into lower-dimensional representation called feature space. By manipulating these vectors, various types of augmentation are obtained. In this regard, we find adversarial training, GAN based augmentation, neural style transfer, meta-learning, and many more.

The goal of this work is to make a tool with the best augmentation method to address STMicroelectronics problem. This is meaningful as upon the semi-supervised classification method already in use the system will grant robustness and higher classification accuracy by quickly populating any new cluster that may arise daily. The method can also grant benefits like reduction of verification time upon classification because there will be no more need to discard newly created classes until more samples come out.

The present chapter is structured as follows: in 4.2 the state of the art of augmentation algorithms will be discussed and how the microprocessor industry has already exploited its potential will be analysed. In 4.3 the method developed and the data used for the tests will be treated. In 4.4 the results of classifications made both with and without augmented images will be displayed. To conclude the chapter, in 4.5 the conclusions and benefits brought to STMicroelectronics will be discussed.

4.2 Related works

The first examples of data augmentation date back to 1998 with handwritten digit classification performed by LeNet-5 [68, 67]. Since then it has spanned through many fields of application such as Deep Learning to medical imaging. The methods for performing augmentation have also evolved over time. At present days we find different augmentation methods adaptable to different types of problems. Recently there has been an evolution of data augmentation methods, passing from standard methods to more modern approaches that harness the power of neural networks to generate new images. In [69] a methodology using blocky artifact with autoencoder to create images for handwritten digits recognition is described and in [70] a method that generates medical images using GAN is proposed, increasing the performance of CNN for liver lesions classification. In [71] a method is proposed that exploits a new contextual loss function capable of preserving context better than traditional approaches. This method is used in augmenting medical images. However, the modern approach to data augmentation has not supplanted the traditional method which is still widely used effectively as highlighted in [72]. Moreover, it happens that the results obtained through a modern approach are not always appreciably higher than a classic approach to justify a considerable use of time by modern models compared to the classic ones. We have much evidence of traditional Data Augmentation in use at present days. In [73] experiments are conducted in which geometric and photometric transformations are compared for use in classification tasks via CNN. In [74] a radial transformation approach is used for augmenting data in classification tasks with a Deep neural network while in [75] a method is described for generating false images by pairing sample images from the dataset. In [76] a method mixes up two known approaches, cropping four different images and patching them into a new image. The method is hence evaluated with a deep CNN. Moreover, tools have recently been made available as python packages which contain a collection of the most famous augmentation methods [77, 78]. Additionally, data augmentation is a technique already in use in semiconductor manufacturing [62], and even in STMicroelectronics it has already been investigated with an ad-hoc procedure to face another classification problem in which the number of defects is known and limited unlike the problem addressed in this work where new classes can be added daily and it hence needs a more general approach [63].



(a) (b) (c) (d) (e)

Figure 4.1: An example of the original images of each class: (a) Image with "center" label, (b) Image with "donut" label, (c) Image with "near-full" label, (c) Image with "edge-ring" label, (c) Image with "scratch" label

4.3 Data and method

The exposed method is based on a mixing images approach as it is described in [79], exploiting the shape of the wafer, circular and almost symmetrical, which lends itself very well to this type of methodology. Dataset used is [14] which proves to be very good to validate the robustness and accuracy of the method given the similarity of the defects present in the database with those in STMicroelectronics dataset. The images were binarized and scaled to 61×61. White pixels represent bad dies while black pixels represent good dies. Descriptors used are PCs. To test the quality and the effectiveness of the images thus created, a k-nn procedure was used. Descriptors and k-nn settings are described in 2.3.1 and 3.3 .

4.3.1 Data

As already mentioned, the method exploits the open WM-811K dataset [14]. 4675 images divided in 5 classes belonging to the dataset have been selected and are divided as follows:

- **2032** labeled as Center;
- **231** labeled as Donut;
- **1491** labeled as Edge-Ring;
- **793** labeled as Scratch;
- **128** labeled as Near-Full;

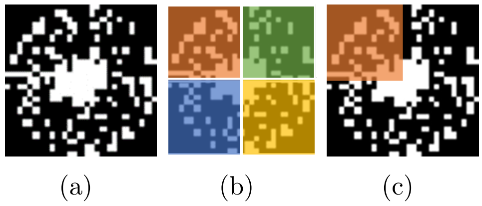


Figure 4.2: First step of Mixed concatenation algorithm: (a) Input image, (b) Input image split in 4 sectors, (c) Random sector selection

A visual representation of example images belonging to each class can be seen in 4.1. By applying the augmentation procedure to each class, a second dataset of augmented images was obtained. In particular, for each label the following number of images was generated:

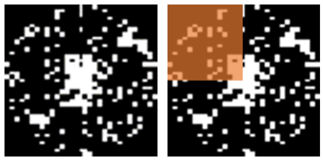
- **1768** as Center;
- **2310** as Donut;
- **1491** as Edge-Ring;
- **1586** as Scratch;
- **1280** labeled as Near-Full;

The final dataset consists of 13110 images of which 4675 original and 8435 augmented.

4.3.2 Method

As already mentioned, the method exploits the circular shape of the silicon wafer. The input image is hence divided into 4 sectors. One of the sectors is then randomly selected by the algorithm. This first step of the procedure is shown in figure 4.2. The algorithm randomly selects a second image to use for the mix within the dataset of images containing the same defect as the input image. The procedure then selects the same sector previously chosen as shown in figure 4.3.

The sector selected in the first wafer is then removed and replaced by the sector of the second image as shown in figure 4.4. The result of the mixing is shown in figure 4.5



(a) (b)

Figure 4.3: Second step of Mixed concatenation algorithm: (a) Second input image, (b) First input image sector selection

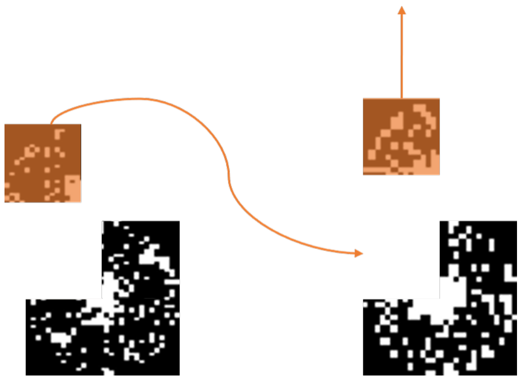


Figure 4.4: Final step of the algorithm. The sector in the first image is replaced with the sector of the second image

Figure 4.5: Mixed image

4.4 Experimental Results

Many experiments have been carried out to test the quality and effectiveness of our method. At first, the experiments were carried out exclusively on the original dataset, dividing the 4675 images into training and test sets randomly. Subsequently, the experiments were carried out with the complete dataset of 13,110 images For each experiment conducted, two versions were provided, one including the percentage of affinity to a given class, since this is a procedure already in use in STMicroelectronics as exposed in the paper [65]. The images that do not reach the affinity threshold will be placed in a new class. In table 4.1 and table 4.2 the confusion matrices of an experiment with both versions are shown. Overall, the classification was performed 10 times. The results with the original data are shown in tables 4.3 and 4.4 while the results with mixed data are shown in tables 4.5 and 4.6. After the augmentation procedures, a general increase in the classification accuracy of the algorithm can be noted. The average results of the algorithm without the affinity percentage can be seen in table 4.7. The average results of the algorithm with the affinity percentage can be seen in table 4.8. As can be seen, the augmentation procedure has significantly increased the accuracy, not only generically but, especially, of the most unbalanced class with donut label. On the other hand, a slight decrease in precision can be noted on the center label after the augmentation procedure, as this has obviously increased the classification uncertainty in favor of other classes.

Table 4.1: Example of confusion matrix, showing the classification outcomes using only original images, with 2305 images as training set and 2370 images as test set.

	Center	**Donut**	**Edge-Ring**	**Scratch**	**Near-Full**
Center	942	3	3	78	0
Donut	0	69	1	36	0
Edge-Ring	6	0	738	28	0
Scratch	9	1	9	378	0
Near-Full	0	0	0	0	69

4.5 Conclusions

The method turns out to be efficient when the starting dataset is composed of binary images with a circular shape such as that of the wafers, thus being particularly suitable for classification problems relating to microelectronics industries such

Table 4.2: Example of confusion matrix, showing the classification outcomes using only original images and affinity percentage, with 2305 images as training set and 2370 images as test set.

	Center	**Donut**	**Edge-Ring**	**Scratch**	**Near-Full**	**New Class**
Center	814	2	0	27	0	183
Donut	0	57	0	16	0	33
Edge-Ring	3	0	704	2	0	63
Scratch	0	0	4	297	0	96
Near-Full	0	0	0	0	69	0

as STMicroelectronics completely replacing the augmentation methods previously analysed and discussed in 3.4. The algorithm was adequate for the generation of fake images that would increase the number of images of the subsampled classes. The effectiveness is found in the appreciable increase in the classification precision, both generic and, as expected, of the classes that presented subsampling problems. The donut class was one of those with few samples, passing from a classification accuracy of **0.659** with just the original images to a precision of **0.923** after the augmentation procedure. The case of the near-full label is different as the particular pattern of the defect does not allow any kind of uncertainty. Therefore, even in the presence of a subsampled class, the classification of this class always has an accuracy of **1**. The developed method is therefore adequate to quickly fill the subsampled classes and to be used in those classification problems related to STMicroelectronics where new classes with few samples may arise every day. A comparison between original images and Augmented images with the same labels is shown in(Figure 4.6). The article containing all the research work done on Data Augmentation is currently waiting for approval by the legal department of STMicroelectronics.

Table 4.3: results of the classification on 4675 images from the original WM-811K dataset, divided randomly and equally into training and test set, on 10 runs without the use of the affinity percentage divided as follows: the first column represents the overall accuracy of the algorithm while the other columns represents the accuracy of each single label.

	%Precision	**%Center**	**%Donut**	**%Edge-Ring**	**%Scratch**	**%Near-Full**
#1	0.926	0.918	0.65	0.955	0.952	1
#2	0.931	0.918	0.685	0.982	0.918	1
#3	0.921	0.894	0.697	0.982	0.926	1
#4	0.933	0.925	0.70	0.982	0.918	1
#5	0.925	0.921	0.654	0.981	0.904	1
#6	0.923	0.901	0.669	0.969	0.955	1
#7	0.920	0.908	0.603	0.979	0.936	1
#8	0.912	0.901	0.6	0.982	0.886	1
#9	0.939	0.938	0.675	0.99	0.907	1
#10	0.929	0.931	0.657	0.972	0.911	1

Table 4.4: results of the classification on 4675 images from the original WM-811K dataset, divided randomly and equally into training and test set, on 10 runs with the use of the affinity percentage divided as follows: the first column represents the overall accuracy of the algorithm while the other columns represents the accuracy of each single label.

	%Precision	**%Center**	**%Donut**	**%Edge-Ring**	**%Scratch**	**%Near-Full**
#1	0.818	0.793	0.537	0.911	0.748	1
#2	0.835	0.834	0.620	0.934	0.682	1
#3	0.835	0.814	0.587	0.959	0.718	1
#4	0.834	0.853	0.603	0.947	0.615	1
#5	0.821	0.857	0.548	0.932	0.584	1
#6	0.838	0.818	0.5	0.913	0.824	1
#7	0.818	0.798	0.496	0.945	0.710	1
#8	0.830	0.839	0.504	0.95	0.647	1
#9	0.827	0.842	0.594	0.947	0.595	1
#10	0.819	0.85	0.552	0.931	0.572	1

Table 4.5: results of the classification on 13110 images both from the original WM-811K dataset and their augmentation, divided randomly and equally into training and test set, on 10 runs without the use of the affinity percentage divided as follows: the first column represents the overall accuracy of the algorithm while the other columns represents the accuracy of each single label.

	%Precision	**%Center**	**%Donut**	**%Edge-Ring**	**%Scratch**	**%Near-Full**
#1	0.95	0.898	0.949	0.987	0.957	1
#2	0.943	0.885	0.922	0.987	0.97	1
#3	0.946	0.90	0.92	0.984	0.962	1
#4	0.949	0.927	0.899	0.988	0.962	1
#5	0.945	0.914	0.915	0.988	0.941	1
#6	0.947	0.903	0.924	0.98	0.967	1
#7	0.946	0.911	0.915	0.986	0.952	1
#8	0.945	0.892	0.921	0.987	0.968	1
#9	0.949	0.911	0.928	0.988	0.952	1
#10	0.95	0.905	0.94	0.984	0.963	1

Table 4.6: results of the classification on 13110 images both from the original WM-811K dataset and their augmentation, divided randomly and equally into training and test set, on 10 runs with the use of the affinity percentage divided as follows: the first column represents the overall accuracy of the algorithm while the other columns represents the accuracy of each single label.

	%Precision	**%Center**	**%Donut**	**%Edge-Ring**	**%Scratch**	**%Near-Full**
#1	0.881	0.828	0.807	0.975	0.849	1
#2	0.883	0.823	0.785	0.971	0.902	1
#3	0.872	0.828	0.765	0.961	0.864	1
#4	0.88	0.871	0.757	0.968	0.847	1
#5	0.866	0.844	0.777	0.974	0.786	1
#6	0.879	0.837	0.779	0.967	0.866	1
#7	0.876	0.847	0.77	0.97	0.846	1
#8	0.886	0.833	0.787	0.963	0.907	1
#9	0.882	0.854	0.792	0.974	0.833	1
#10	0.886	0.845	0.808	0.956	0.880	1

Table 4.7: average precision of the algorithm with original and mixed data, without affinity percentage, in general and for each single label.

	Original data	**Mixed data**
General accuracy	0.925	**0.947**
Center	**0.915**	0.904
Donut	0.659	**0.923**
Edge-Ring	0.977	**0.985**
Scratch	0.921	**0.959**
Near-Full	1	1

Table 4.8: average precision of the algorithm with original and mixed data, with affinity percentage, in general and for each single label.

	Original data	**Mixed data**
General accuracy	0.827	**0.879**
Center	0.829	**0.841**
Donut	0.554	**0.782**
Edge-Ring	0.936	**0.967**
Scratch	0.669	**0.858**
Near-Full	1	1

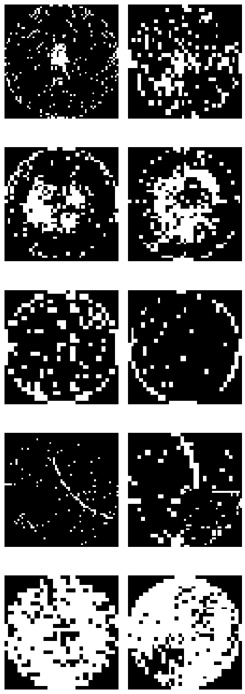


Figure 4.6: Original images (left column) compared with augmented images with the same labels (right column).

Chapter 5

Low yield Apriori searching tool

5.1 Introduction

Data mining is a multidisciplinary research field with many fields of application such as, but not limited to, machine learning and statistics. Data mining was the answer to the huge growth of data that is processed and stored in databases every day. Indeed, in 1991 the amount of information was already estimated to double every 20 months [80]. Growing databases contain valuable information but human ability, of course, was unable to analyse this information alone. The purpose of data mining is, hence, the search for knowledge in databases [81]. Mining association rules is, hence, one of the many purposes in data mining. It was mainly introduced in the problem known as basket analysis [82, 83] or, in other words, the need to discover hidden connections in purchase transactions. Basket data type could be both items bought together at the same time or over a while. In general, to formalize the problem, given a set of items I where:

$$I = I_1, I_2, I_3, ..., I_n \tag{5.1}$$

and given a set of transactions T and a transaction $t \in T$ it is defined:

$$t[k] = 1 \tag{5.2}$$

if $i_k \in t$ and it is defined

$$t[k] = 0 \tag{5.3}$$

if $i_k \notin t$.

An association rule is defined as:

$$X \Rightarrow I_j \tag{5.4}$$

where X is a set of items in I and I_j is a single item in I.

As mentioned before, association rules mining is a widely used method for finding relationships between items within a set of transactions. One of the first applications is found in the search for frequent associations within a set of transactions to generate frequent association rules, which are often generated in basket analysis.

A different approach, on the other hand, can be found in the search for those infrequent associations which, sometimes, can be more significant than frequent associations in a set of transactions. This is called rare association rules mining [84].

The manufacturing process in STMicroelectronics is particular and complex and corruption of the finished product could occur in the presence of errors or malfunctions within the production flow. STmicroelectronics' request is to find hidden associations between harmful events, which can be machine alarms, process interruptions, or human errors, which generate low yield products. The basic assumption for ST technicians is that frequent events may not be significant to explain low yield products. Rare events, on the other hand, could be a symptom of momentary situations potentially harmful if prolonged over time, such as the malfunction of a machine that had worked effectively up to the moment in which the event occurred. Is, hence, their need to quickly search for connections between rare events and low yield products to act promptly on the cause of the event itself and even stop the production before the process is completed.

To meet the needs of STMicroelectronics, many algorithms already known in literature were analysed. Among the most promising we find the Apriori algorithm [85, 86, 87], the FP-growth algorithm [88, 89] and their derivations [90, 91, 92, 93, 94]. For the present case, it was decided to use the Apriori algorithm for its simplicity and effectiveness. Not only; several studies have already been completed with success to modify the algorithm and make it able to detect associations between rare events rather than frequent ones [95], associations that, as mentioned before, STMicroelectronics needs to find.

The present chapter is structured as follows: in 5.2 the state of the art of association rules learning and how it has already been used in semiconductor manufacturing will be discussed. In 5.3 the method developed and the data used for the tests will be treated. In 5.4 the results of our method for rare associations mining will be displayed. To conclude the chapter, in 5.5 the conclusions and benefits brought to STMicroelectronics by our rare association mining method will be discussed.

5.2 Background and Related works

As mentioned in 5.1 basket analysis is a branch of enormous scientific importance, as highlighted by plenty of articles published since association rules mining began to be addressed by the scientific community [82, 96]. Different approaches to solve the association rules mining problem have, hence, been analysed and, over time, it has been an improvement and strengthening of the observed methods, thus reflecting the enormous importance that this problem has over the world [97, 84, 83, 98, 95]. Apriori was one of the first algorithms within the associative rules mining landscape [85] and it was, therefore, one of the first answers to the problem of basket analysis. The main issue of the algorithm was its computational slowness, due to its huge database scanning. Many studies have been carried out over time, both to improve the effectiveness of the algorithm [99, 91, 86, 92, 93] and its field of action, orienting research not only towards frequent itemsets but also towards the search for rare associations [100, 101, 95].

On the other hand, another algorithm appeared in the association rules mining landscape: the FP-Growth algorithm. The algorithm works very differently from Apriori. Instead of performing multiple searches within the database, FP-Growth treats the database as a tree, which can save considerable amounts of memory for storing the transactions, making it, in fact, at that time a faster algorithm than Apriori [88]. Over time, this algorithm has also undergone improvements [90, 94]. Nowadays, both algorithms are currently employed to solve data mining problems [89].

Furthermore, even in business sectors such as semiconductor manufacturing, there is much evidence of the application of these methodologies for solving problems that have many affinities with association rules mining and basket analysis in general [102]. First of all, the need for knowledge discovery and correlations between parameters has already been highlighted by STMicroelectronics in the past years [103].

One of the first examples of the use of data mining techniques dates back to 2001 [104] when a rule-structuring algorithm was proposed for semiconductor manufacturing industries. Another example can be found, in more modern times, in 2017 [105], where it is exposed a Big-Data-based monitoring system in semiconductor manufacturing which also includes the use of association rule learning techniques. In particular, they use the FP-Growth algorithm for low yield cause identification, considering the manufacturing history of a given product as a transaction.

As already mentioned, given a set of items I and a set of transactions T, an association rule is a logical connection between two elements of I highlighted by their presence, more or less frequent, in the transactions T. To formalize the model we can adduce the following considerations: I is a set of items;

$$I = i_1, i_2, ..., i_n \tag{5.5}$$

T is a set of transactions;

$$T = t_1, t_2, ..., t_m \tag{5.6}$$

R is an association rule;

$$R = X \Rightarrow Y : X, Y \subseteq I \tag{5.7}$$

X is called antecedent while Y is called consequent and it is read X implies Y.

S is the support of an item and it is defined as follows:

$$S(X) = \frac{|t \in T; X \subseteq t|}{|T|} \tag{5.8}$$

The support of an item shows how often that item can be found in the transaction set.

C is the confidence of a given rule R and it is defined as follows:

$$C(R) = C(X \Rightarrow Y) = \frac{S(X \cup Y)}{S(X)} \tag{5.9}$$

The confidence of a rule shows the possibility of having Y given X within T.

Given the premises, the support for a rule is the fraction between the union of both consequents and antecedents and the total transactions in T. On the other hand, the confidence of a rule is the fraction between the union of both consequents and antecedents and the total transactions in T having X as antecedent and it is a measure of the rule strength. The definitions just mentioned are needed to highlight the functioning of the Apriori algorithm. Apriori is, hence, the algorithm chosen to solve association rules problems in STMicroelectronics. Unlike FP-Growth, in fact, best suits the needs of STMicroelectronics thus solving more effectively the problems related to rare itemset mining.

As already mentioned, there are many evidence on the effectiveness of the Apriori algorithm to solve problems related to rare itemset mining. The standard Apriori algorithm works as mentioned below [85]:

- **Input:** a transaction dataset;
- **Large 1-itemset:** the algorithm counts item occurrences to determine large 1-itemsets;
- **Associations making:** the algorithm generates $k+1$ length candidate itemsets from large itemsets of length k;
- **Candidate pruning:** the algorithm removes those candidate itemsets having subset of length K that are not large;
- **Support check:** the algorithm checks for support of each candidate itemset;
- **Candidate pruning:** the algorithm once again removes all those candidate itemsets that are considered small;
- **Output:** the algorithm generates all the association rules meeting the minimum support and minimum confidence threshold;

Of course, to be considered large, an itemset has to meet the minimum support threshold.

5.3 Data and Method

The exposed method is based on the Apriori algorithm to solve rare itemset mining issues. The Apriori algorithm itself was, hence, modified to meet STMicroelectronics needs. Data used for testing the algorithm were made of production events related to STMicroelectronics that occurred over the past years. The dataset has been modified to be functional for use with Apriori algorithm.

5.3.1 Data

Apriori algorithm takes as input a transaction dataset where each element of the transaction is treated as an integer or a boolean as already explained in equations 5.2 and 5.3. STMicroelectronics dataset is made of production events that occurred during the fabrication process of silicon wafers. Each silicon wafer belongs to a production lot. An example of how the dataset is structured can, therefore, be represented as shown in table 5.1. The first column contains the wafers to which each transaction refers. The other columns, except for the last one, contain production alarms that occurred during production flow. The last column contains the yield of the wafer to which the transaction relates. The yield consists of a percentage from 0 to 100. Thus, STMicroelectronics technicians empirically set a threshold below which a wafer could be considered bad. Below a threshold of 80%, the yield is set to 1, otherwise, it is set to 0.

5.3.2 Method

The developed method is based on the Apriori algorithm for finding rare association rules. In the beginning, it was decided to test the basic version of the Apriori algorithm to verify its validity and effectiveness. A support dataset was used to look for frequent association rules that lead to low yield for wafers. The Apriori algorithm in its basic version had already proved effective according to the technicians of

Table 5.1: Example of the structure of the STMicroelectronics dataset with production events. The first column contains all wafers and their respective lots. From the second to the penultimate column are the production events, or alarms. Any alarm whose value is 0 means that the alarm did not occur during that wafer's production flow. On the other hand, any alarm whose value is 1 means that the alarm occurred during that wafer's production flow. The last column contains the yield of that wafer. A yield value of 0 means that the yield was good enough while a yield value of 1 means that, following the negative events that have occurred, the yield of that wafer is negative and must therefore be discarded.

Lot_Id/Wafer_Id	**Alarm_1**	**Alarm_2**	**Alarm_3**	**...**	**Alarm_n**	**Yield**
Lot_1/Wafer_1	0	1	0	...	1	0
Lot_1/Wafer_2	1	0	1	...	1	0
Lot_2/Wafer_1	1	0	1	...	1	1
Lot_2/Wafer_2	1	1	1	...	0	1
...	...	...	...	...	...	...
Lot_i/Wafer_j	1	1	0	...	1	1

STMicroelectronics who were immediately enthusiastic about the algorithm's effectiveness and its potential. To meet the needs highlighted by STMicroelectronics, the algorithm has undergone several improvements. Above all, the execution time had to be improved. Indeed it exceeded two hours when we had to look for association rules of length equal to three and, therefore, in the form:

$$R = X(i_0, i_1) \Rightarrow Y(i_n) : i_0, i_1, i_n \subseteq I \tag{5.10}$$

i_0, i_1 and i_n are, thus, distinct elements of the dataset.

5.3.3 Candidate Itemset

The first improvement and, perhaps, the most important one that Apriori underwent is the Candidate Itemset reduction. In other words, the Apriori algorithm generates, at each scan of the database, the relative candidate itemset. In general, for the K^{th}

iteration, the algorithm generates C_k candidate itemset. C_k itemset is the superset of C_{k-1} itemset. An example of candidate itemset generation is shown in table 5.2.

STMicroelectronics technicians empirically set the chain limit for association rules to 3. According to their experience, it is thus not necessary to search for more than three connections as a low production yield should be sought on the frequency of a single event over time or, at most, on the frequency of an event and a contributing cause.

As already explained, the first experiments with the basic version of Apriori took a long time to be performed as it took 2 hours for a chain of 3 connections. To reduce the execution time it was therefore decided to reduce the candidate itemset. The solution was found in the definition of the problem itself, as STMicroelectronics was only interested in association rules having one or two alarms as antecedent and the low production yield as a consequent. It must be remembered that low production yield was included in the dataset as an event, as explained in table 5.1. Yield equal to 1 means that the wafer of that specific transaction has a low yield, while a wafer having yield equal to 0 has a positive yield.

It is, hence, easy to modify the algorithm so that, once it reaches the last desired iteration, it would generate connections only and exclusively with the yield element. An example of the flow of the standard Apriori algorithm can be seen in tables 5.3, 5.4, 5.5. First of all, it generate the first candidate itemset C_1 5.3. As a second step it generate the candidate itemset C_2, a superset of C_1 5.4. Finally it generates the last candidate itemset C_3, superset of C_2 5.5. An example of the flow of our modified Apriori algorithm can be seen in tables 5.6, 5.7, 5.8. The first candidate itemset generation, C_1, in table 5.6, is equal to the base version shown in 5.3. The second step is a bit different. The algorithm generates the candidate itemset C_2, shown in table 5.7, combining the elements in C_1 but without the redundant pairs. Finally, the third step is very different. The algorithm generates the candidate itemset C_3, shown in table 5.8, combining the pairs in C_2 only with *yield* element thus reducing enormously the execution time by removing a large number of unnecessary searches in the dataset.

5.3.4 Association rules filtering

A further level of improvement has been obtained by adding a method for filtering the associations found by the algorithm. The idea was born from the considerations of STMicroelectronics technicians. The idea is that a certain rule should be significant if the number of wafers of a certain lot, affected by the rule, should be almost equal or, at least, similar to the total number of bad wafers of that lot. Given an association rule R:

$$R = X \Rightarrow Y \tag{5.11}$$

the association rule filter R_f can be defined:

$$R_f = \frac{W_{lot} \in R}{W_{lot}} \simeq 1 \tag{5.12}$$

W_{lot} is the number of bad wafers belonging to a single lot t. $W_{lot} \in R$ is the number of bad wafers belonging to the same lot t that make the rule R true.

This rule indicates a symptom of an event that could have damaged many wafers belonging to a single production lot if all or most of the bad wafers belonging to that lot have suffered the same event. This makes the rule appreciable, otherwise, the same rule is discarded.

5.4 Experimental Results

Experiments have been conducted on a dataset belonging to STMicroelectronics which included alarms data gathered for nine months. **For privacy reasons, the dataset used cannot be disclosed.**

As already mentioned, the Apriori basic algorithm has been modified to meet the needs expressed by the technicians of STMicroelectronics. The functioning of the basic algorithm will be explained below:

- **Rules Definition:** the algorithm generates a set of rules R_n;
- **Confidence filter:** the algorithm filters the generated rules R_n according to the confidence C setting and generates the rules set $R_i \subseteq R_n$ where $C(R_i) \geq C|(i = 1, 2, ...n)$

The modified Apriori algorithm version is shown below:

- **Rules Definition:** the algorithm generates a set of rules R_n;
- **Confidence filter:** the algorithm filters the generated rules R_n according to the confidence C setting and generates the rules set $R_i \subseteq R_n$ where $C(R_i) \geq C|(i = 1, 2, ...n)$
- **Alarm filter:** the algorithm filters the generated rules R_i according to the alarm filter and generates the rules set $R_j \subseteq R_i$ where $alarmfilter \simeq 1$

The parameterization of the Apriori algorithm was set by STMicroelectronics technicians and is shown below:

- **Support:** $S(I) \leq 0.1\%$ (Rare associations mining);
- **Maximum Chains:** $chain \leq 3$ $R = (alarm_1, alarm_2) \Rightarrow (yield)$;
- **Confidence:** $C(R) \geq 0.85\%$;

With the addition of the constraint on the connections, limiting the consequents to the yield, execution times have been significantly reduced as it can be seen in figure 5.1.

After the experiments, STMicroelectronics technicians estimated the algorithm success rate is 71.6% with 2625 generated rules.

5.5 Conclusions

The method, after the necessary improvements in terms of execution speed, turns out to be suitable to meet the needs of STMicroelectronics. Forcing the algorithm to search only for those connections having yield as consequent element proves to be fundamental, as the method takes a few minutes to execute, unlike the basic version which took two hours for chains of length 3 as it can be seen in figure 5.1.

The preliminary results turn out to be promising as a success rate of 71.6% proves to be appreciable. Further analysis is needed to improve the algorithm. An appreciable improvement on the algorithm performance should be achieved with a different parameterization, by changing the values of support and confidence.

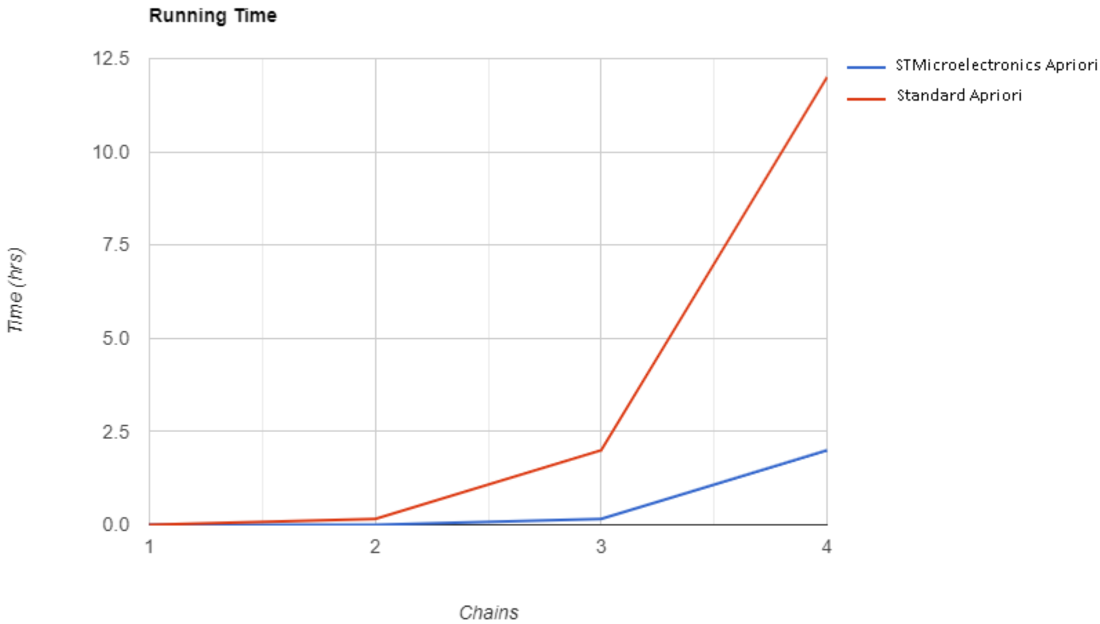


Figure 5.1: Running time for both versions of Apriori Algorithm with 1, 2, 3, 4 chains

Furthermore, it could be useful to try different values for the ratio of total bad wafers of a given lot with the bad wafers of that lot after a given rule.

At present, the method proves to be a good starting point for tackling the problem of association rules learning. STMicroelectronics will continue the research with this modified version of the Apriori algorithm as a starting point.

The article containing all the research work done on Association Rules Learning is currently waiting for approval by the legal department of STMicroelectronics.

Table 5.2: An example of candidate itemsets of a chain of length 2. In the first column, the candidate itemset C_1 is composed of the individual elements of the dataset that fall within the requirements defined by the threshold of the *support* variable. The second column defines the candidate itemset C_2 composed of the pairs generated between the elements of C_1

C_1 **itemset**	C_2 **itemset**
i_1	$(i_1,i_2),(i_1,i_4)$
i_2	$(i_2,i_1),\ (i_2,i_3),(i_2,i_4)$
i_3	$(i_3,i_2),\ (i_3,i_5)$
i_4	$(i_4,i_1),\ (i_4,i_2),\ (i_4,i_5)$
i_5	$(i_5,i_3),(i_5,i_4)$

Table 5.3: An example of candidate itemsets of a chain of length 1. The candidate itemset C_1 is composed of the individual elements of the dataset that fall within the requirements defined by the threshold of the *support* variable. In this example there are 4 alarms, i_1, i_2, i_3, i_4 and, finally, the *yield*

C_1 **itemset**
i_1
i_2
i_3
i_4
yield

Table 5.4: An example of candidate itemsets of a chain of length 2. The candidate itemset C_2 is composed of the combinations of each element in C_1 of table 5.3 that fall within the requirements defined by the threshold of the *support* variable. In this example, for convenience, we assume that all the generated pairs fall within the limits defined by the threshold

C_2 **itemset**
$(i_1,i_2),(i_1,i_3),(i_1,i_4),(i_1,yield)$
$(i_2,i_1),(i_2,i_3),(i_2,i_4),(i_2,yield)$
$(i_3,i_1),(i_3,i_2),(i_3,i_4),(i_3,yield)$
$(i_4,i_1),(i_4,i_2),(i_4,i_3),(i_4,yield)$
$(yield,i_1),(yield,i_2),(yield,i_3),(yield,i_4)$

Table 5.5: An example of candidate itemsets of a chain of length 3. The candidate itemset C_3 is composed of the combinations of each element in C_2 of table 5.4 with each single element of C_1 of table 5.3 that fall within the requirements defined by the threshold of the *support* variable. In this example, for convenience, we assume that all the generated chains fall within the limits defined by the threshold

C_3 **itemset**
(i_1,i_2,i_3), (i_1,i_2,i_4), $(i_1,i_2,yield)$
(i_1,i_3,i_2), (i_1,i_3,i_4), $(i_1,i_3,yield)$
(i_1,i_4,i_2), (i_1,i_4,i_3), $(i_1,i_4,yield)$
$(i_1,yield,i_2)$, $(i_1,yield,i_3)$, $(i_1,yield,i_4)$
(i_2,i_1,i_3), (i_2,i_1,i_4), $(i_2,i_1,yield)$
(i_2,i_3,i_1), (i_2,i_3,i_4), $(i_2,i_3,yield)$
(i_2,i_4,i_1), (i_2,i_4,i_3), $(i_2,i_4,yield)$
$(i_2,yield,i_1)$, $(i_2,yield,i_3)$, $(i_2,yield,i_4)$
(i_3,i_1,i_2), (i_3,i_1,i_4), $(i_3,i_1,yield)$
(i_3,i_2,i_1), (i_3,i_2,i_4), $(i_3,i_2,yield)$
(i_3,i_4,i_1), (i_3,i_2,i_2), $(i_3,i_4,yield)$
$(i_3,yield,i_1)$, $(i_3,yield,i_2)$, $(i_3,yield,i_4)$
(i_4,i_1,i_2), (i_4,i_1,i_3), $(i_4,i_1,yield)$
(i_4,i_2,i_1), (i_4,i_2,i_3), $(i_4,i_2,yield)$
(i_4,i_3,i_1), (i_4,i_3,i_2), $(i_4,i_3,yield)$
$(i_4,yield,i_1)$, $(i_4,yield,i_2)$, $(i_4,yield,i_3)$
$(yield,i_1,i_2)$, $(yield,i_1,i_3)$, $(yield,i_1,i_4)$
$(yield,i_2,i_1)$, $(yield,i_2,i_3)$, $(yield,i_2,i_4)$
$(yield,i_3,i_1)$, $(yield,i_3,i_2)$, $(yield,i_3,i_4)$
$(yield,i_4,i_1)$, $(yield,i_4,i_2)$, $(yield,i_4,i_3)$

Table 5.6: An example of candidate itemsets of a chain of length 1 generated by our modified Apriori Algorithm. The candidate itemset C_1 is composed of the individual elements of the dataset that fall within the requirements defined by the threshold of the *support* variable. In this example there are 4 alarms, i_1, i_2, i_3, i_4 and, finally, the *yield*

C_1 **itemset**
i_1
i_2
i_3
i_4
$yield$

Table 5.7: An example of candidate itemsets of a chain of length 2 generated by our modified Apriori Algorithm. The candidate itemset C_2 is composed of the combinations of each element in C_1 of table 5.6 that fall within the requirements defined by the threshold of the *support* variable, without the redundant pairs generated by the basic Apriori version. In this example, for convenience, we assume that all the generated pairs fall within the limits defined by the threshold

C_2 **itemset**
$(i_1,i_2),(i_1,i_3),(i_1,i_4),(i_1,yield)$
$(i_2,i_3),(i_2,i_4),(i_2,yield)$
$(i_3,i_4),(i_3,yield)$
$(i_4,yield)$

Table 5.8: An example of candidate itemsets of a chain of length 3 generated by our modified Apriori Algorithm. The candidate itemset C_3 is composed of the combinations of each element in C_2 of table 5.7 with only *yield* element of C_1 of table 5.6. In this example, for convenience, we assume that all the generated chains fall within the limits defined by the threshold

C_3 **itemset**
$(i_1,i_2,yield)$
$(i_1,i_3,yield)$
$(i_1,i_4,yield)$
$(i_2,i_3,yield)$
$(i_2,i_4,yield)$
$(i_3,i_4,yield)$

Chapter 6

Final considerations

With this dissertation, we wanted to demonstrate the contribution made to the growth of STMicroelectronics through the research and studies conducted during the Ph.D. training period. In the work presented in chapters 2 and 3, we focused on a very specific kind of data from semiconductor manufacturing called Electrical Wafer Sorting (EWS) maps. These images are generated during the wafer testing phase performed in semiconductor device fabrication. We assumed to handle binary EWS maps, where white pixels identify failed dies, while black pixels represent the good ones. Usually, yield detractors are identified by specific and characteristic patterns, named anomalies signatures. These patterns are useful for investigating the root causes that could be, for instance, related to an equipment component failure, a drifting process, or an integration of processes [1]. Unfortunately, it is not possible to resort to a predefined and fixed set of labels for the observed defects in the images. New kinds of defects may arise daily in an unpredictable way and in this case new labels have to be produced.

In the paper [65], we presented a new semi supervised approach for classifying anomalies signatures in EWS maps, by combining an unsupervised approach using a hierarchical clustering algorithm to create the starting Knowledge base, and a supervised one through a classifier trained by leveraging clustering phase. The knowledge base represents our core knowledge about the possible anomaly signatures (i.e., the number of clusters) known until each daily update. We therefore dynamically proceeded to test our clustering procedure on the incrementing dataset. Indeed, our dataset can be daily increased, and the classifier is dynamically updated considering possible newly created clusters. The workflow of our solution can be resumed in:

- Daily arrival of EWS maps;

- Anomalies signatures classification;
- Possible creation of new clusters;

We compared several clustering and classification techniques. We found that aggregative hierarchical clustering leveraging Principal Components computed through the Principal Component Analysis can be a robust clustering method. Then, we trained a Convolutional Neural Network with ResNet-18 architecture, reaching performance comparable with other state-of-the-art techniques. We remark that our method does not rely on any labeled dataset and can be daily updated, unlike comparative literature. Our dataset is skewed, a common characteristic in real use-case industrial scenario. Moreover, we proposed a method that was proved to be rotation invariant.

The goal of this work was to create a tool to make as automatic as possible the recognition of wafer anomalies signatures. This is meaningful as upon classification the industrial system can be able to automatically choose (or at least suggest) either to discard a wafer or to ship it to the customer. The proposed method can also grant benefits like reduction of wafer test results review time, or improvement of processes, yield, quality, and reliability of production using the information obtained during clustering process. As a sideline, we have analysed various data augmentation techniques and we have developed an augmentation algorithm that is as generic as possible by exploiting the circular shape of the wafer. The method splits each image into four sectors and replaces one of the four sectors with the same sector taken from another image with the same defect. The method has proven to be effective for rapidly populating classes with few samples.

To conclude we have developed an algorithm for rare association mining to search for hidden associations between production alarms and low yield products. The developed method is based on a modified version of the Apriori algorithm and it proved to be a good starting point for pushing research in the direction we have highlighted.

As future works, we planned to investigate the performance of other CNN architectures. We also designed a comparison study with a two-fold purpose: consolidate outcomes shown in chapters 2 and 3 employing the WM-811K dataset, and exploring the existence of any correlation with test phases before the EWS (e.g., relatively to Wafer Defect Maps - WDM). Furthermore, it was decided to push the research towards rare association mining to evaluate different settings for the developed Apriori algorithm and modify the custom filters that have been created to increase the precision of the method. STMicroelectronics has shown appreciation for the work done about Semi-Supervised classification of EWS signatures and, given the enormous strategic advantage found and the factory's internal impact of the method has decided to put the proposed architecture in their production flow.

Appendices

The articles published on secondary activities carried out during the PhD period will be listed below. The first article shows a tool developed on Unity game engine, Curve Design Studio, to generate and manipulate bézier curves [106]. The second article shows a tool developed on Unity as well to easily create point and click serious games oriented to cultural heritage: Point To Culture [107].

Appendix A

Curve Design Studio: Bézier curve integrated tool for video game development

A.1 Introduction

Computer graphics tries to represent reality in a virtual and simulated environment. This led the researchers to propose new algorithms for realistic image synthesis. However, the realism we have at the present day could not have been achieved without the exploitation of specific mathematical instruments [108, 109].

A mathematical concept widely used in computer graphics, especially in video games industry, is parametric curve [110, 111]. The use of parametric curves is needed in many aspects of the video game development process like path definition, car routes, 3D modelling and so on. In the early 1960s, Peter Bézier began researching a better way to represent curves and surfaces that could have been useful for automotive engineers. He was familiar with cubic curves and bicubic surfaces researched by Ferguson and Coons [112]. However, these methods had drawbacks, especially when shapes alteration and control was required. Bézier Curves were proposed to face this problem and became part of the UNISURF system, used by the French automobile manufacturer Renault to design the surfaces of many of their products.

Due to its simple geometric construction, Bézier curves are widely used in many fields, and, with the growth of computer graphics applications, it was an opportunity to introduce it as a tool for vector drawing software [113]. This allows to realise high precision drawings with little effort. Nowadays, computer graphics programs and design systems provides tools for Bézier Curve manipulation, so that they are used for many practical applications. They are even employed for designing the shape of letters in fonts and for many other graphical and practical applications [114, 115]. In the past decades, the video game industry focused on the use of Bézier curves and their derivatives for the representation of 3D graphical assets in order to hide defects that emerge from a polygonal realisation [110]. They have also been used in animations and time driven events [116, 117]. We have evidence of curves employment in game engines like `Unity3D`, but they are usually coded for specific purpose [118, 119]; alternatively, curves are used in third-part software [120, 121, 122] for creating 3D objects, like roads and other curve shaped object which are subsequently imported in game engines. It is possible to generate curves in other engines such as `Unreal engine` but only with coding, without the possibility of customization already present in other tools. We have evidence of a conceptual use of Bézier curves for terrain development [123] and also an application developed on `Unity` for the construction of terrains by using curves [124]. Other software improve the `Unity particle system` by exploiting the Bézier curves [125]. Nevertheless, the programmer is usually forced to program their curve manually. This lack of tools and algorithms for using curves and exploiting their full potential has motivated our work.

In this paper we propose "Curve Design Studio", a novel tool embedded in `Unity3D Game Engine` for visual representation of a Bézier curve and for its application in many tasks of video game development process. The use of a plugin embedded in a game engine capable of manipulating a curve may save video games companies financial resources and time. Indeed, they may not be constrained to use other computer graphic applications for those precision tasks. The proposed tool implements smart algorithms to recreate a Bézier curve in the `Unity3D virtual environment` without the need to rely on external software. The proposed tool allows to assign custom parameters to the curve and the related points and real time changing. To prove the validity of the proposed work, at the end of the paper we present a custom solution for the movement of an object along the curve. The rest of the paper is organised as follows: in Section A.2 we state the problem and analyse previous works. In Section A.3 we introduce Curve Design Studio. In Section A.4

we analyse quality and efficiency of the proposed plugin through the use of a custom script: FollowPath. It uses the proposed tool for moving a 3D object in the virtual environment. In Section A.5 we report the results and our conclusions.

A.2 Statement of the problem and previous works

Our work is motivated by the lack of `Unity3D` tools for using Bézier curve in many aspects of a video game development process. This is a well-known problem to the community of game developers. Over the years, the community has created many auxiliary tools capable of manipulating a Bézier curve in order to meet these lacks. Unfortunately, each tool addresses a specific problem. Video game developers are hence forced to use multiple software thus increasing project complexity and development time. Some tools to create Bézier curves are hence present within the `Unity3D asset store`. Among them: *Bézier Curve Editor, Bézier Master, Spline Mesh, BG Curve* [126, 127, 128, 129]. However they all focus on a limited videogame development aspects and do not offer a comprehensive overview of what could be done by using curves; moreover, they have a too slow learning curve and are not recommended for those without programming skills. Bézier Curve Editor permits an easy creation and modification of Bézier splines. Such spline can be used as a path for game objects and as animation curve. The toolkit also has APIs for getting various spline info, but they are minimal. Bézier Curve Editor has a big limitation: it is not possible to edit or generate a spline in Game. SplineMesh is a powerful tool with an intuitive editor and excellent performance, great for creating the following content:

- A Bézier spline. (However, it is not possible to retrieve much mathematical information);
- 3D Mesh extrusion, bending or cloning through a spline;
- 2D Mesh extrusion along the spline.

SplineMesh does not have the flexibility of Curve Design Studio from a mathematical point of view but it remains, however, an excellent tool. BG Curve is one of the most complete curve-related asset package. First of all, it is possible to generate and modify a Bézier spline both in real time and in the scene view and a lot of mathematical information can be obtained from the generated spline, thanks to several dedicated APIs. BG Curve has many additional features: the creation of a mesh, the movement of game objects, the creation of 3D or 2D colliders, the cursors and so on. However, given the complexity of the mentioned plugin, Curve Design Studio turns out to be more practical and easier to use.

The proposed tool addresses several problems related to the use of a curve in a video game development process such as:

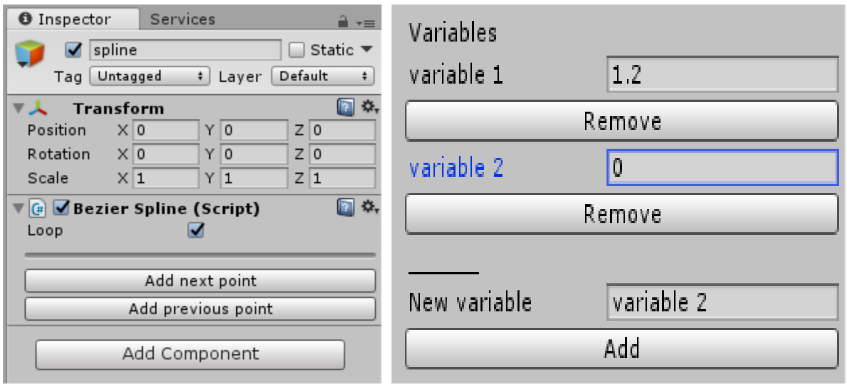
- animating an object along a path;
- create paths and terrain along a curve;
- instantiating objects on the points of a curve;
- extrude an object along a curve;
- creating complex behaviours for artificial intelligence agents (patrolling, car driving, etc).

Moreover, it aims to provide a framework which can be easily extended.

A.3 Method

`Unity3D engine` is an environment which integrates a game engine provided by Unity Technologies. `Unity` is typically employed to produce digital games for different platform, such as PC's, consoles, mobile devices and websites. It allows to handle 3D model and other kinds of assets, such as materials, lights, images, and videos. `Unity` gives the possibility to encode algorithms in two different programming languages: C# and JavaScript. Although `Unity` is often used for digital game development, it could be employed for generic purpose application related to 3D scenes a models. The main feature of `Unity` is the convenient way to manage multimedia data, the user-friendly development GUI and the multi-platform builder [130].

In this work we propose a novel tool called Curve Design Studio as plugin for `Unity3D`. Our tool implements the concept of Bézier curve in a three-dimensional



(a) (b)

Figure A.1: (a) GameObject with Bézier Spline script attached showing customization elements within `Unity editor`; (b) Custom variables for anchor points;

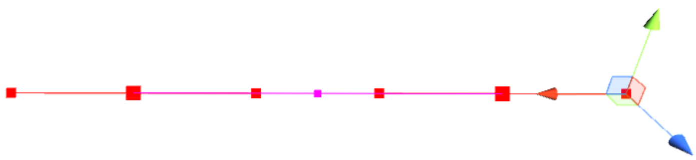


Figure A.2: Standard linear Bézier curve where anchor and control points lie along the same line.

space, giving developers the chance to use the curve without binding it to a specific problem. It can be used to address multiple issues which usually occur during the development of a visual application. The algorithm allows the construction of a Bézier curve through its anchor and control points. An example of a standard linear Bézier curve in the `Unity virtual environment`, where all points are aligned, is shown in Fig. A.2. This curve is automatically created in the `Unity editor` when the script Bézier Spline is attached inside an empty gameobject previously inserted in the `Unity editor` scene, as it is shown in Fig. A.1(a). The user can manipulate the curve through the control points, represented by the smaller squares. Indeed, selecting these points the `Unity` visual manipulation system will appear, represented by the presence of the three Cartesian axes. The bigger squares represents the anchor points.

Hence, it is possible to manipulate these points in the three-dimensional space

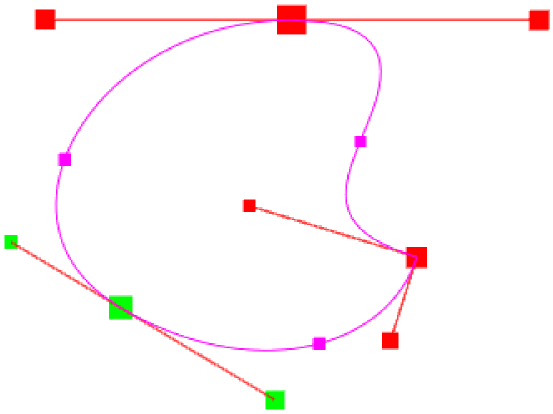


Figure A.3: Composite Bézier curve

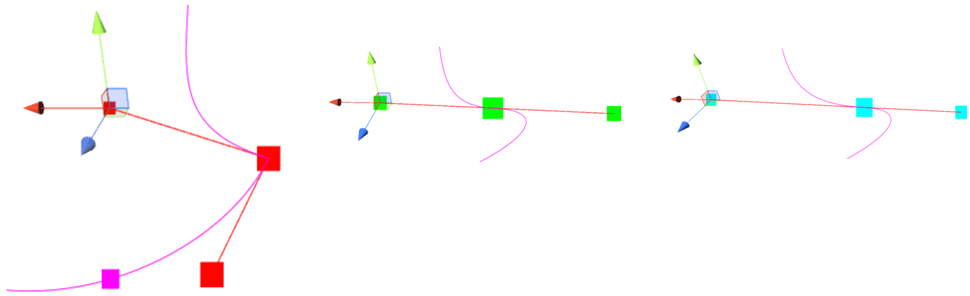
and to assign some custom variables to them. Moreover, the algorithm extends the visual editor of `Unity` and provides useful tools for curve manipulation even in game-mode. What is permitted in this case is:

- to close the spline with a check box called loop;
- to add other points to the curve;
- to move a selected point;
- to delete a selected point;
- to restore a deleted point.

A more complex curve, realised with the help of the extended visual editor, is shown in Fig. A.3.

Additionally, by moving the points, it is possible to change their own behaviour with an attribute called *Style*. Three styles for each point can be selected:

- Free style. The control points are unbounded. (see Fig. A.4(a))
- Mirror style. Control points will form a line with the anchor point and must be at the same distance from it. (see Fig. A.4(b))
- Align style. Control points will form a line with the anchor point but they can have different distance from it. (see Fig. A.4(c))



(a) (b) (c)

Figure A.4: (a) Free style; (b) Mirror style; (c) Align style;

In addition, as mentioned before, it is possible to assign parameters to any point in the spline. When selecting one of the available points, the `Unity editor` will show a control window where the user can add variables and remove them as shown in Fig. A.1(b) To remarks the advantages of the proposed tool, we show an example application creating by using Curve Design Studio: Followpath. It shows how a 3D object could be easily moved, in the three-dimensional environment, along the Bézier curve.

Moreover, it shows how to manipulate its parameters, i.e. the scale factor and the custom variables assigned to the curve points.

A.3.1 Preliminary

Here, we provide mathematical notions of a Bézier curve, and how to reach its graphical representation through the De Casteljau's algorithm.

Bézier curve representation

Given three points, A, B, C, so that line in $\overline{AB}$ is tangent to the curve at A and line $\overline{BC}$ is tangent to C, the curve begins at A and ends at C. For any ratio u_i, where $0 \leq u_i \leq 1$, the points D and E are constructed so that

$$\frac{\overline{AD}}{\overline{AB}} = \frac{\overline{BE}}{\overline{BC}} = u_i \tag{A.1}$$

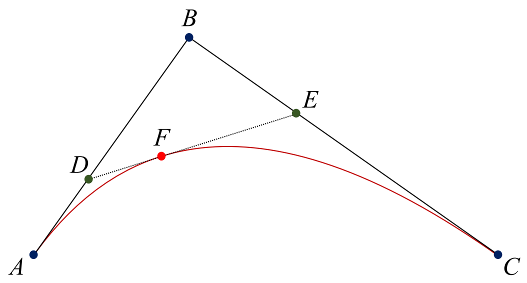


Figure A.5: Second-degree Bézier curve

On $\overline{DE}$, F is constructed so that

$$\frac{\overline{DF}}{\overline{DE}} = u_i \tag{A.2}$$

Hence, the point F belongs to the curve. For other values of u_i, a series of points are produced on a Bézier curve (see Fig. A.5) [114]. *These points are called control points. The first and last point on the curve are called anchor points.*

De Casteljau's algorithm

A Bézier curve Be of grade n, with structural points $P_0 ... P_n$ could be written in Bernstein form as follows:

$$Be(T) = \sum_{i=0}^{n} P_i b_{i,n}(t) \tag{A.3}$$

where $0 \leq t \leq 1$ and b is Bernstein polynomial [131]

$$b_{i,n}(t) = \binom{n}{i} (1-t)^{n-i} t^i \tag{A.4}$$

Nevertheless, it is not recommend directly using the equation (A.3) to generate a curve in a digital system, as it is not numerically stable. This pushes researchers to find a more stable strategy to draw a Bézier curve in such systems. De Casteljau's algorithm is an effective, recursive and stable method to evaluate Bernstein polynomials or Bézier curve [132]. It is robust methods which gives insight into Bézier curve behaviour and leads to important operations on the curves.

According to De Casteljau's algorithm, one can evaluate a Bézier curve at t_0 with the following recurrence formula

$$P_i^{(0)} := P_i \quad i = 0, ..., n \tag{A.5}$$

$$P_i^{(j)} := P_i^{(j-1)}(1 - t_0) + P_{i+1}^{(j-1)} t_0 \quad i = 0, ..., n \quad j = 1, ..., n \tag{A.6}$$

$$Be(t_0) = P_0^{(n)} \tag{A.7}$$

Hence, the recursive definition led the following equation, which is equal to the Bernstein form:

$$Be(t) = \sum_{i=0}^{n} \binom{n}{i} (1 - t)^{n-i} t^i P_i \tag{A.8}$$

where $P_0, P_1 ... P_n$ are control points [133, 134].

Using De Casteljau's algorithm we represent the Bézier curves in the proposed plugin. In this work, we implement cubic Bézier curves, which are uniquely defined by 4 control points.

A.3.2 Curve Design Studio implementation

Here we provide a technical description of the the proposed tool.

Algorithm 1 Bézier

```
procedure GetPoint
    Input: P0,P1,P2,P3, t
    Output: p4
    t ← clamp 0 ≤ t ≤ 1
    i ← 1-t
    p4 ← i^3*P0+3*i^2*t*P1+3*i*t^2*P2+t^3*P3
```

The Cubic Bézier curve is defined into an abstract class called Bézier. The procedure is reported in Algorithm 1. Given four points $P_0 ... P_3$ and a percentage t, the method outputs the point in the curve corresponding to that percentage. To store the spatial information for each point in the curve, three lists of vectors were implemented. Two of them are used for the control points, since we have two control points for each anchor point and one list is used for the anchor points. Each point has an index.

Algorithm 2 GetNextPointIndex

Input: i
Output: j
if loop **then**
    $i \leftarrow (i{+}1)\%n\_points$
    return i
if $i == n\_points - 1$ **then return** -1
return $++i$

Given an index for a point in the curve, the procedure for taking the next index is reported in Algorithm 2. n_points is the number of points in the list while *loop* identifies a closed curve. If a loop is found, the index gets back to 0 when it gets to the last index in the list. If not and it found the last index it returns -1. Otherwise it just returns the next index.

A.3.3 Point style

A list of *int* named *Style* is used to store the value of the style that can be assigned to each point. In this case, the parameter can get the following values:

- $0 = Free$
- $1 = Mirror$
- $2 = Align$

Algorithm 3 is used when a control point is moved or you have to change its style. In the case, given an index i, and a control point of index i and a $flag$, which identifies the first or the second control point, the algorithm moves the first or the second control point, specified by the parameter $flag$, taking the position of the other control point as a reference to ensure the correctness of the specified style. The method MoveToward moves the specified control point *pt* from its original position toward the position specified by the second input parameter *controlpoint* of the distance specified by the last input parameter *Distance*.

Algorithm 3 FixPtStyle

Input:i,$flag$
$style \leftarrow Style[i]$
if $style == 0$ **then return**
if $flag == 0$ **then**
  if $style == 1$ **then**
    $P_{2,i} \leftarrow$ MoveToward$(pt_i$,$P_{1,i}$,
    -Distance$(pt_i$,$P_{1,i}))$
  if $style == 2$ **then**
    $P_{2,i} \leftarrow$ MoveToward$(pt_i$,$P_{1,i}$,
    -Distance$(pt_i$,$P_{2,i}))$
if $flag == 1$ **then**
  if $style == 1$ **then**
    $P_{1,i} \leftarrow$ MoveToward$(pt_i$,$P_{2,i}$,
    -Distance$(pt_i$,$P_{2,i}))$
  if $style == 2$ **then**
    $P_{1,i} \leftarrow$ MoveToward$(pt_i$,$P_{2,i}$,
    -Distance$(pt_i$,$P_{1,i}))$

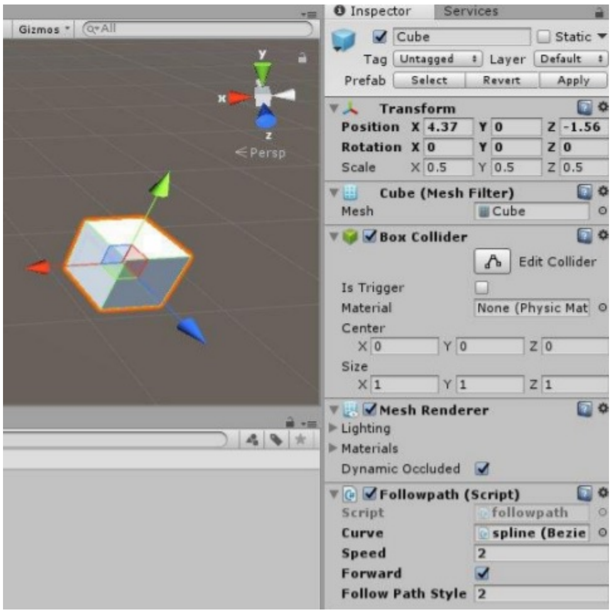


Figure A.6: Followpath configuration. To use it properly a spline must be attached in the appropriate field of the interface, assign a speed, direction and type of movement

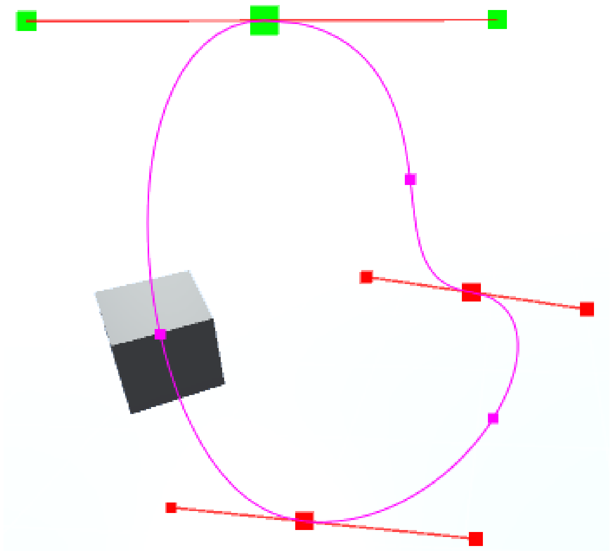


Figure A.7: Cube moving along the curve. After followpath script configuration, the cube will move along the curve drawn in purple. If variables have been added to the control points, represented by the purple squares, the cube will assume different behaviors (for example it will be scaled if there is a scale variable) once the points are reached. The other squares are anchor points; they can be used to manipulate the shape of the curve. Their colors are the standard `Unity` colors and indicate the direction of the selected axis

A.3.4 Custom Variables

As it was mentioned before, it is possible to assign one or more variables to each point in the curve. The same variable, therefore, can assume different values depending on the anchor point in which it is located. this peculiarity is very important as, in the proposed plugin, there is an algorithm that allows to obtain the interpolated value of this variable in any point of the curve between two anchor points.

A.4 Case of study

To test the quality of the analysed tool a custom script for `Unity` called FollowPath was realised. It allows to move a three dimensional object along a curve made with Curve Design Studio. Moreover it permits to modify the scale of that object exploiting some custom parameters. Thanks to Curve Design Studio this script implementation is an easy task. The only thing that is needed is a primitive three dimensional object. `Unity` provides lot of built-in objects making it easy to add a Cube in the virtual environment. After adding the cube, the script FollowPath must be linked to the object itself. `Unity` shows us some properties that we have to customize, as it is shown in Fig A.6. The first thing we have to do is to link the Bézier curve to the curve variable. The object, while in play mode, can move along the linked curve, as can be seen in Fig A.7. The speed variable controls the speed at which the object can move. The check box forward force the cube to orientate its z axis toward the curve direction. Last parameter is the follow path style and controls the type of movement the cube will have. A value of 2 means that the cube will move in a loop. Adding a scale variable on the points of the curve will force the cube to change its scale factor according to the value of that scale parameter.

A.5 Conclusion

We have presented Curve Design Studio, a plugin embedded in `Unity3D` game engine which permits the realisation of Bézier curves easily. The plugin comes with a complete suit of tools to manipulate and customise the curve and its points. Unlike the tools already present in the market, Curve Design Studio does not address any specific issue but provides the developers with a powerful tool which can be personalised and adapted to the needs that may arise during a video game development process. The versatility and strength of Curve Design Studio have been shown with FollowPath, a custom script that uses Curve Design Studio and its features to solve some issues like the movement of a three dimensional object or its scale factor manipulation. Its ease of use and its versatility make it a good tool to embed in `Unity3D` game engine replacing actual tools for curve manipulation. As future works we are organizing tests with selected groups of users with various expertise, ranging from simple users to experienced developers. In this way we will have a more accurate overview of the effectiveness and simplicity of the proposed tool.

Appendix B

Point to Culture: a point to click framework for Serious Games in Cultural Heritage

B.1 Introduction

The videogame industry has spread widely in recent years and the purpose of videogames has changed over time. The ever-increasing requests to bring the game closer to educational aspects led the emergence of Serious Game. Hence the videogame turned from an entertainment tool to a mean for cultural and educational purpose where the entertaining power of the videogame is not the primary goal but plays a support role.

Health care, military simulations, school education are just few examples of effective usage of the Serious Game. Lately we find an ever increasing use of Serious Games for Cultural Heritage as they promise a greater engagement and interests from players, providing a powerful mean for knowledge dissemination, as well as an effective means for digitisation and preservation of cultural sites [135, 136, 137]. The interest for Serious Games applied to Cultural Heritage has grown with the introduction of newer technologies such as 3D scan, virtual and augmented reality [138] yet without abandoning the idea of developing simple story driven Serious Games without any other support device. Hence, this led the scientific community to improve both technologies and design for Serious Games applied in Cultural Heritage [139, 140] providing guidelines for better understanding what the community needs, in terms of development tools and game design. In this paper we propose Point to

Culture a novel tool embedded in Unity3D Game Engine that makes it simpler and more intuitive the development process of simple 2D point-to-click games. The tool implements smart algorithms that facilitate the realisation of common features in a point-to-click game, such as but not limited to:

- `items`: Use and combine pickable objects.
- `inventory`: Inventory system.
- `explorable scenes`: Switching between interactive scenes.
- `puzzles`: Create puzzles like numerical combinations or use the correct object over an interactive item.
- `dialogue system`: Create interactive narrative experience.
- `cutscenes`: Creating static cutscenes by combining two or more images.

To prove the validity of the proposed tool we present a case study of a custom Serious Game set in the famous Benedictine Monastery of Catania [141, 142]. The game resumes the well-known format of escape rooms. The player, gaining knowledge from the Monastery, has to use the information learned to escape and win the game. The contribution of this paper can be summarised in:

- A novel Unity plugin to support the development of point-to-click digital games.
- A case-study Serious Game set in the Benedictine Monastery of Catania, developed by using the proposed plugin.

The remainder of this paper is organised as follows: Section B.2 describes all the previous works done for Serious Games development and the technology improvement for Cultural Heritage Serious Games; Section B.3 details the framework and its components; in Section B.4 we present a case study game developed with the proposed tool and by remarking its cultural relevance; Section B.5 concludes the paper.

B.2 Related Works

The approach to Serious Game development has changed and evolved with technological changes. Virtual and augmented reality provided powerful tools for Serious Games, specially for Cultural Heritage. Augmented reality allows an effective in-site gamification and is extensively employed for Cultural Heritage study [143, 144]. On the other hand, Virtual reality gives the opportunity to virtually visit any location from home. However, this does not make obsolete classical Serious Games, specially if they are story driven, have to teach something about the cultural site and need a player rather than a virtual visitor. Indeed, such games have been implemented and have proven their effectiveness as stated in [145].

Several studies have been conducted on best practices and models for better implementing efficient Serious Games [146, 147, 148, 149].There are many valid approaches proposed during the past years [150], but what the scientific community seems to agree on is the format for developing Serious Games for Cultural Heritage. To be effective they should be story driven and task based, like puzzle games or adventure games [151, 152, 153]. Furthermore, there are many game engines on the market, even professional ones. However, Unity and Unreal turn out to be the most exploited [154, 155]. Unity is appreciated for its ease of use compared to Unreal, which is more suitable for professional video games. Even though there are some adventure makers plugin in the Unity Asset Store they are designed for more complex games providing a harsh learning curve for Serious Games designers.

At best of our knowledge there are few references to frameworks dedicated to task based Serious Games like Sketch'ndo [149] which is not a plugin for a Game Engine. It is built on top of Blender Graphical engine and it works with Blender logic connections providing limited flexibility. Differently, our plugin provides a powerful tool for adventures and point-to-click games. Adventure Creator [156] is a plugin for Unity that allows the development of traditional adventure games. It is certainly a valid tool for the creation of point-to-click games, also taking advantage of a visual programming through the presence of a node editor. It is a very large framework that contains many functions, such as a dialogue system and a visual development tool and it is certainly the most suitable tool for large productions but its complexity and the large amount of documentation necessary for its use make it a tool not suitable for simpler games such as, for example, Serious games. In fact, the time that should be spent learning how to use the tool would be excessive compared to the minimum complexity required for a serious game.

B.3 Proposed framework

Unity3D engine is an environment which integrates a game engine provided by Unity Technologies. Unity is typically employed to produce digital games for different platform, such as PC, consoles, mobile devices and websites. The main feature of Unity is the convenient way to manage multimedia data, the user-friendly development GUI and the multi-platform builder [130].

In this work we propose a novel tool called Point to Culture as plugin for Unity3D. Our framework provides all the necessary tools for easily develop 2D point-to-click video-games that are well suited as Cultural Heritage Serious Games. It can be used to easily address multiple common tasks which usually occur during the development of this type of games. Usually, the development process of a serious game can be summed up in three phases [157]:

- `Brainstorming`: The theoretical conception of the video-game with mechanics and dynamics.
- `Prototype development`: Multi-phase development of prototypes up to the realisation of the finished product.
- `Validation`: Validate the effectiveness of the game.

The proposed plugin fits into the prototype development phase, providing all the necessary tools for rapid prototyping, leaving developers the opportunity to focus more on the other phases thus reducing development times and costs. The plugin appears in the form of a toolbar element providing a drop down menu. Each menu item opens a dedicated section where it is possible to build almost every game element without a single line of code.

One of the most common aspects of 2D point-to-click games is the presence of multiple items and the need for the programmer to insert complex interactions such as the possibility of combining two or more objects and use them with each others or together with other scene elements. For example, the user could use a key to open a chest or a door or he could combine a hammer head with a handle to obtain a hammer to use against a wall. After selecting the relevant menu item, a window will open, as it is shown in Fig. B.1 where the programmer can insert the sprite that provides the exterior appearance of the item and whether or not it is obtained by the union of other items. In this case, the programmer simply selects the items providing the combination, or the recipe, to get the composed item. To place them on the game scene, the programmer selects the dedicated menu element, selects the item to place, and puts it on the scene with just a click.

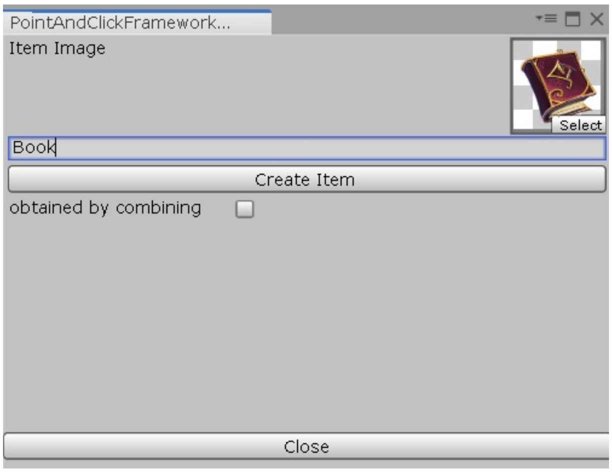


Figure B.1: Create item window

B.3.1 Inventory

Another important aspect of 2D point-to-click games is the presence of the player inventory, a virtual storage where to insert items picked during the game. To create it, as it is shown in Fig. B.2, the programmer needs to select the sprites representing the frame for the items, the background of the inventory and the items already in the inventory at the beginning of the game, if any.

B.3.2 Combination Puzzle

The plugin permits to create a numerical combination by just selecting the relevant item on the menu. After selecting the background, up and down arrow sprite and

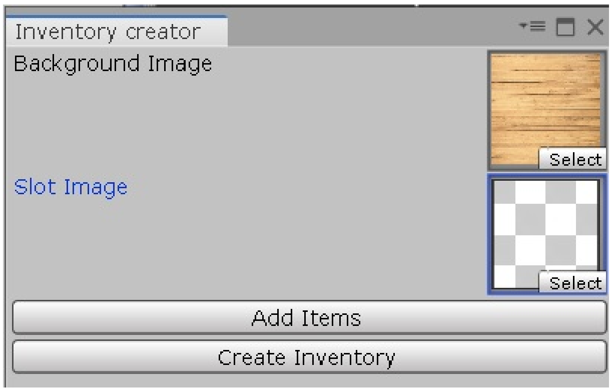


Figure B.2: Inventory window

the digit background, a screen with a single digit will be created. The programmer could then add more digits by selecting the relevant button. The screen created could be linked to any puzzles inside the game, either a chest with a numerical padlock or a door with a numeric keypad.

B.3.3 Background image and Cutscenes

By Selecting the relevant menu item, the programmer could create many static images providing the game scenarios which will be automatically connected through the presence of directional arrows that could be customized, also in appearance, and placed around the background. It is also possible to create simple cutscenes by just selecting a group of static images and the time lapse to change between them.

B.3.4 Conditions and reactions

To add complex interactions between game elements we provide a powerful tool easy to use. Interactive, Reaction collection and Condition collection represents three scripts providing complex interactions through the mechanism of action and reaction. The Interactive script provides interactivity to a game element. The Condition collection provides a list of allowed events for that specific element, linking each Condition to a specific Reaction. Interactive script allows an item in the game scene to be interactive. The programmer could add one or more behaviour to that specific item, in addition to a standard reaction which will be performed if no conditions are met. An interactive behaviour consists of the combination of a list of conditions and a list of reactions.

Conditions

The Condition collection script allows the programmer to add to a specific interactive item a list of scripted conditions that may or may not occur. It is also possible, for a programmer, to extend such script adding more conditions to expand the plugin. The conditions already included are the followings:

- **Combination condition** is the condition that allows the insertion of numerical puzzles, such as a numerical padlock to open a door or a chest. The Combination condition links a specific combination puzzle previously created to a specific interactive item. The Combination condition allows to insert the correct combination number for a specific combination puzzle. If the player inserts the right combination in the game, the condition is met and a reaction will be triggered.
- **Selected inventory item condition:** this is the condition that allows to add the interaction between a specific item in the inventory with an item in the game scene, for example a key for a chest or a key-card for a door.
- **State condition** is the condition that allows an item in the game scene to get a state, which is a boolean, which will trigger a reaction if the status is reached. For example a chest that gets open or a door unlocked.

Reactions

The reaction collection script allows the programmer to add several effects in response to any condition met. Following the reactions already scripted and their behaviour are explained:

- **Active gameObject reaction:** when a condition is met activate a gameObject inside the scene.
- **Change scene reaction:** when a condition is met load another scene.

- **Change sprite reaction:** when a condition is met, change the sprite of an object inside the scene. This is used, for example, to change the appearance of a chest whether it gets opened.
- **Change state reaction:** when a condition is met, use it to change the state of an item in the scene, for example change the state of a door from closed to open.
- **Dialog reaction:** when a condition is met, triggers a dialog. It is used, for example, to trigger conversations with non playable characters.
- **Open combination reaction:** open the combination interface.
- **Take item reaction:** when a condition is met, puts an item in the player inventory.

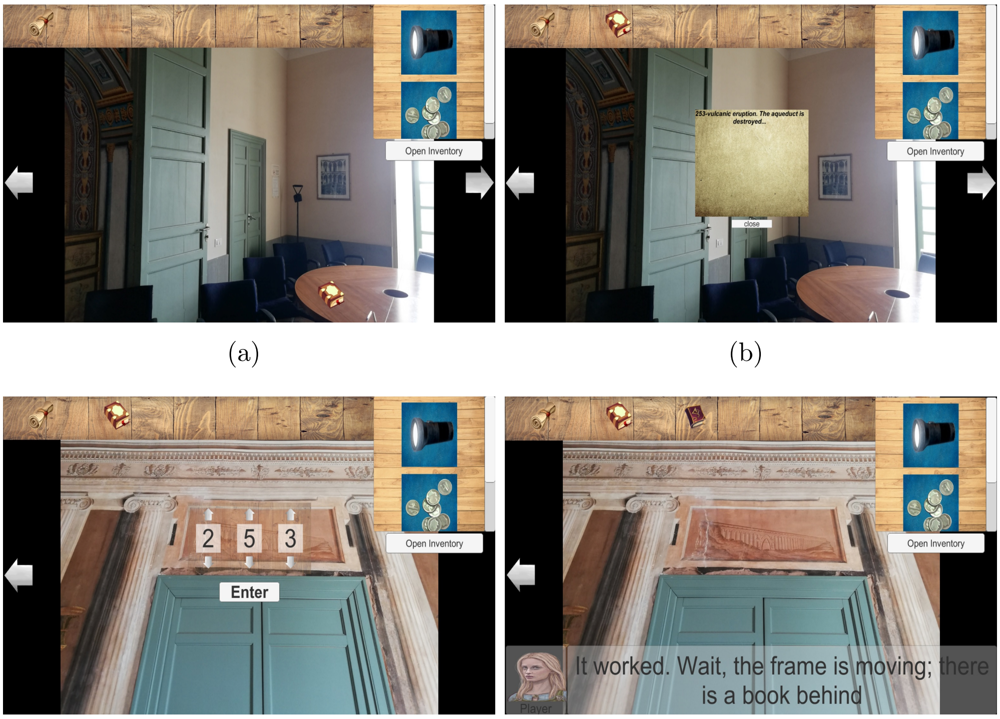
B.4 Escape the Benedictine Monastery

As a case-study it was chosen to develop an escape room game set in the former abbot's room in Benedictine Monastery of Catania. The choice of the Monastery as a game environment was led by the enormous importance of the Monastery and the Historical relevance of its environments. Furthermore, its structure is well suited to be brought back to a game like the escape room.

B.4.1 Case-study game

The player can independently visit the abbot's room and look for clues to leave the room. To win the game, the player will become aware of some historical facts related to various cultural elements that can be found in the room. In this case, as it is shown in Fig. B.3, a short sequence taken from the demo, the player will find objects scattered around the room containing important historical facts to end the game.

In the sequence mentioned above, in Fig. B.3(a) and Fig. B.3(b) the player, clicking on a book on the table will have access to historical information concerning the year of destruction of the Catania aqueduct.



(a) (b)

(c) (d)

Figure B.3: (a) Book item; (b) Book informations; (c) Knowledge verification; (d) Reward;

In Fig. B.3(c) the player is facing the representation of the Catania aqueduct. An interact script attached to the aqueduct representation will trigger a message window. At the end of the dialogue, the player will be informed of the presence of a numeric keypad. The player will need not only to remember what information concerning the aqueduct he was able to find but, above all, to understand what is the correct information to use in this situation.

In Fig. B.3(d) after entering the date of the destruction of the aqueduct, another dialog will be triggered. The player will be informed that he has entered the correct solution and will get the reward. In the present case he will find a book, appeared in the upper area of the inventory, containing other historical facts about the monastery with which he has to solve other puzzles to win the game. The purpose of a serious game is to entertain the user, improving his specific knowledge of the topic covered in the game. Indeed, the player will be required to explore the history of the monastery as the knowledge of many historical details will prove essential to win the game, mixing, hence, the playful details of a video game, for example puzzles and riddles, with the historical details necessary for the educational purpose, according to the design principles of serious games [153].

B.5 Conclusions

The proposed plugin represents a valid tool for simple 2D point-to-click games oriented to Cultural Heritage. Its ease of use and the vastness of standard functions provided makes it one of the ideal tools to easily develop Cultural Heritage Serious Games in a short time. Furthermore, its modularity makes it easily extendable by a programmer, making it adaptable to any needs. As future works, we are planning to massively employ the proposed tool for developing a set of complete Serious Games to support Cultural Heritage dissemination and education. The simplicity of use of the plugin allows the creation of point-to-click video games, with any setting, photorealistic or drawn, in a short time, leaving the developer the possibility to focus more on the educational and narrative aspects rather than programming and development. With this plugin, indeed, we want to provide a development standard where it is possible to produce point-to-click serious games with any setting by changing only the graphics by interacting directly with the plugin interface. Furthermore we are organizing tests with selected group to evaluate both the plugin with more or less experienced developer groups, and the game with groups of players.

B.6 Acknowledgement

We would like to thank Francesco Mannino for the collaboration in finding the images and historical references about the Benedictine Monastery.

Bibliography

[1] G. S. May and C. J. Spanos. *Fundamentals of semiconductor manufacturing and process control.* Wiley Online Library, 2006.

[2] M. Tvaronaviciene, K. Razminiene, and L. Piccinetti. “Aproaches towards cluster analysis”. In: *Economics & sociology* 8.1 (2015), p. 19.

[3] J. Han, M. Kamber, and A. K. Tung. “Spatial clustering methods in data mining”. In: *Geographic data mining and knowledge discovery* (2001), pp. 188–217.

[4] M. Wang, Z. B. Abrams, S. M. Kornblau, and K. R. Coombes. “Thresher: determining the number of clusters while removing outliers”. In: *BMC bioinformatics* 19.1 (2018), p. 9.

[5] M.-F. Balcan, Y. Liang, and P. Gupta. “Robust hierarchical clustering”. In: *The Journal of Machine Learning Research* 15.1 (2014), pp. 3831–3871.

[6] D. Bryant and V. Berry. “A structured family of clustering and tree construction methods”. In: *Advances in Applied Mathematics* 27.4 (2001), pp. 705–732.

[7] M. R. Ackermann, J. Blömer, D. Kuntze, and C. Sohler. “Analysis of agglomerative clustering”. In: *Algorithmica* 69.1 (2014), pp. 184–215.

[8] X. J. Zhu. *Semi-supervised learning literature survey.* Tech. rep. University of Wisconsin-Madison Department of Computer Sciences, 2005.

[9] X. Zhu and A. Goldberg. *Introduction to semi-supervised learning.* Morgan & Claypool Publishers, 2009.

[10] W. Zhang, X. Li, S. Saxena, A. Strojwas, and R. Rutenbar. “Automatic clustering of wafer spatial signatures”. In: *Proceedings of the 50th Annual Design Automation Conference.* ACM. 2013, p. 71.

[11] M.-J. Wu, J.-S. R. Jang, and J.-L. Chen. “Wafer map failure pattern recognition and similarity ranking for large-scale data sets”. In: *IEEE Transactions on Semiconductor Manufacturing* 28.1 (2014), pp. 1–12.

[12] M. Saqlain, B. Jargalsaikhan, and J. Y. Lee. “A Voting Ensemble Classifier for Wafer Map Defect Patterns Identification in Semiconductor Manufacturing”. In: *IEEE Transactions on Semiconductor Manufacturing* 32.2 (2019), pp. 171–182.

[13] C. H. Jin, H. J. Na, M. Piao, G. Pok, and K. H. Ryu. “A Novel DBSCAN-based Defect Pattern Detection and Classification Framework for Wafer Bin Map”. In: *IEEE Transactions on Semiconductor Manufacturing* (2019).

[14] WM-811K. *Wafer Maps Image dataset available on Kaggle*. https://www.kaggle.com/qingyi/wm811k-wafer-map. Online; last visited 09/19. 2018.

[15] R. Di Bella, D. Carrera, B. Rossi, P. Fragneto, and G. Boracchi. “Wafer Defect Map Classification Using Sparse Convolutional Networks”. In: *International Conference on Image Analysis and Processing*. Springer. 2019, pp. 125–136.

[16] R. K. Blashfield and M. S. Aldenderfer. “The literature on cluster analysis”. In: *Multivariate Behavioral Research* 13.3 (1978), pp. 271–295.

[17] J. Scoltock. “A survey of the literature of cluster analysis”. In: *The Computer Journal* 25.1 (1982), pp. 130–134.

[18] A. Serra and R. Tagliaferri. “Unsupervised learning: clustering”. In: *Encyclopedia of Bioinformatics and Computational Biology; Elsevier: Amsterdam, The Netherlands* (2019), pp. 350–357.

[19] D. Xu and Y. Tian. “A comprehensive survey of clustering algorithms”. In: *Annals of Data Science* 2.2 (2015), pp. 165–193.

[20] J. MacQueen et al. “Some methods for classification and analysis of multivariate observations”. In: *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*. Vol. 1. 14. Oakland, CA, USA. 1967, pp. 281–297.

[21] N Rajalingam and K Ranjini. “Hierarchical clustering algorithm-a comparative study”. In: *International Journal of Computer Applications* 19.3 (2011), pp. 42–46.

[22] E. Cesario, A. Vinci, and X. Zhu. "Hierarchical Clustering of Spatial Urban Data". In: *International Conference on Numerical Computations: Theory and Algorithms*. Springer. 2019, pp. 223–231.

[23] S. Guha and N. Mishra. "Clustering data streams". In: *Data stream management*. Springer, 2016, pp. 169–187.

[24] L. McInnes, J. Healy, and S. Astels. "hdbscan: Hierarchical density based clustering". In: *Journal of Open Source Software* 2.11 (2017), p. 205.

[25] S. S. Gleason, K. W. Tobin Jr, and T. P. Karnowski. "Spatial signature analysis of semiconductor defects". In: *Solid State Technology* 39.7 (1996), pp. 127–133.

[26] Y.-S. Jeong, S.-J. Kim, and M. K. Jeong. "Automatic identification of defect patterns in semiconductor wafer maps using spatial correlogram and dynamic time warping". In: *IEEE Transactions on Semiconductor manufacturing* 21.4 (2008), pp. 625–637.

[27] K. W. Tobin Jr, S. S. Gleason, T. P. Karnowski, S. L. Cohen, and F. Lakhani. "Automatic classification of spatial signatures on semiconductor wafer maps". In: *Metrology, Inspection, and Process Control for Microlithography XI*. Vol. 3050. International Society for Optics and Photonics. 1997, pp. 434–444.

[28] K. W. Tobin Jr, S. S. Gleason, T. P. Karnowski, and S. L. Cohen. "Feature analysis and classification of manufacturing signatures based on semiconductor wafer maps". In: *Machine Vision Applications in Industrial Inspection V*. Vol. 3029. International Society for Optics and Photonics. 1997, pp. 14–25.

[29] K. W. Tobin Jr, S. S. Gleason, T. P. Karnowski, and H. Sari-Sarraf. *Automated defect spatial signature analysis for semiconductor manufacturing process*. US Patent 5,982,920. 1999.

[30] F. Lin, C.-K. Hsu, and K.-T. Cheng. "Learning from production test data: Correlation exploration and feature engineering". In: *2014 IEEE 23rd Asian Test Symposium*. IEEE. 2014, pp. 236–241.

[31] M. B. Alawieh, F. Wang, and X. Li. "Identifying wafer-level systematic failure patterns via unsupervised learning". In: *IEEE transactions on computer-aided design of integrated circuits and systems* 37.4 (2017), pp. 832–844.

[32] F. Murtagh and P. Contreras. "Algorithms for hierarchical clustering: an overview, II". In: *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 7.6 (2017), e1219.

[33] C. K. Reddy and B. Vinzamuri. "A survey of partitional and hierarchical clustering algorithms". In: *Data Clustering*. Chapman and Hall/CRC, 2018, pp. 87–110.

[34] H. Singh, F. Lakhani, P. Proctor, and A. Kazakoff. "Defect-data management system at SEMATECH". In: *Solid State Technology* 38.12 (1995), pp. 75–79.

[35] A. Hadid, G. Zhao, T. Ahonen, and M. Pietikäinen. "Face analysis using local binary patterns". In: *Handbook of Texture Analysis*. World Scientific, 2008, pp. 347–373.

[36] S. P. Mishra, U. Sarkar, S. Taraphder, S. Datta, D. P. Swain, R. Saikhom, S. Panda, and M. Laishram. "Multivariate statistical data analysis-principal component analysis (PCA)". In: *International Journal of Livestock Research* 7.5 (2017), pp. 60–78.

[37] M. Turk and A. Pentland. "Eigenfaces for recognition". In: *Journal of cognitive neuroscience* 3.1 (1991), pp. 71–86.

[38] Z. Li and M. de Rijke. "The impact of linkage methods in hierarchical clustering for active learning to rank". In: *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM. 2017, pp. 941–944.

[39] F. Murtagh and P. Legendre. "Ward's hierarchical agglomerative clustering method: which algorithms implement Ward's criterion?" In: *Journal of classification* 31.3 (2014), pp. 274–295.

[40] S. Xu, X. Qiao, L. Zhu, Y. Zhang, C. Xue, and L. Li. "Reviews on determining the number of clusters". In: *Applied Mathematics and Information Sciences* 10.4 (2016), pp. 1493–1512.

[41] T. Thinsungnoena, N. Kaoungkub, P. Durongdumronchaib, K. Kerdprasopb, and N. Kerdprasopb. "The clustering validity with silhouette and sum of squared errors". In: *learning* 3 (2015), p. 7.

[42] K. He, X. Zhang, S. Ren, and J. Sun. "Deep Residual Learning for Image Recognition". In: *CoRR* abs/1512.03385 (2015). arXiv: 1512.03385. URL: http://arxiv.org/abs/1512.03385.

[43] T. G. Dietterich. "Machine learning for sequential data: A review". In: *Joint IAPR international workshops on statistical techniques in pattern recognition (SPR) and structural and syntactic pattern recognition (SSPR)*. Springer. 2002, pp. 15–30.

[44] R. Caruana and A. Niculescu-Mizil. "An empirical comparison of supervised learning algorithms". In: *Proceedings of the 23rd international conference on Machine learning*. 2006, pp. 161–168.

[45] T. Cover and P. Hart. "Nearest neighbor pattern classification". In: *IEEE transactions on information theory* 13.1 (1967), pp. 21–27.

[46] C. M. Bishop et al. *Neural networks for pattern recognition*. Oxford university press, 1995.

[47] E. B. Baum and F. Wilczek. "Supervised learning of probability distributions by neural networks". In: *Neural information processing systems*. 1988, pp. 52–61.

[48] P. Hart. "The condensed nearest neighbor rule (Corresp.)" In: *IEEE transactions on information theory* 14.3 (1968), pp. 515–516.

[49] J. Gou, H. Ma, W. Ou, S. Zeng, Y. Rao, and H. Yang. "A generalized mean distance-based k-nearest neighbor classifier". In: *Expert Systems with Applications* 115 (2019), pp. 356–372.

[50] P. Li, J. Gou, and H. Yang. "The distance-weighted k-nearest centroid neighbor classification". In: *J. Intell. Inf. Hiding Multimedia Sig. Process* 8.3 (2017), pp. 611–622.

[51] Y. Zhang, G. Cao, B. Wang, and X. Li. "A novel ensemble method for k-nearest neighbor". In: *Pattern Recognition* 85 (2019), pp. 13–25.

[52] B Yegnanarayana. "Artificial neural networks for pattern recognition". In: *Sadhana* 19.2 (1994), pp. 189–238.

[53] A. Miller, B. Blott, et al. "Review of neural network applications in medical imaging and signal processing". In: *Medical and Biological Engineering and Computing* 30.5 (1992), pp. 449–464.

[54] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. "Backpropagation applied to handwritten zip code recognition". In: *Neural computation* 1.4 (1989), pp. 541–551.

[55] Y. LeCun, Y. Bengio, et al. "Convolutional networks for images, speech, and time series". In: *The handbook of brain theory and neural networks* 3361.10 (1995), p. 1995.

[56] C. Dong, C. C. Loy, K. He, and X. Tang. "Learning a deep convolutional network for image super-resolution". In: *European conference on computer vision*. Springer. 2014, pp. 184–199.

[57] F. F. Ting, Y. J. Tan, and K. S. Sim. "Convolutional neural network improvement for breast cancer classification". In: *Expert Systems with Applications* 120 (2019), pp. 103–115.

[58] C. Zhao, B. Ni, J. Zhang, Q. Zhao, W. Zhang, and Q. Tian. "Variational convolutional neural network pruning". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2019, pp. 2780–2789.

[59] D.-X. Zhou. "Universality of deep convolutional neural networks". In: *Applied and computational harmonic analysis* 48.2 (2020), pp. 787–794.

[60] T. Pan, J. Yang, W. Wu, Z. Dong, and B. O. Ayinde. "Artificial intelligent matching for scratches of semiconductor wafers based on a K-NN algorithm". In: *Surface Topography: Metrology and Properties* 7.2 (2019), p. 025012.

[61] H. Zhang, P. Wang, X. Gao, H. Gao, and Y. Qi. "Automated Fault Detection Using Convolutional Auto Encoder and k Nearest Neighbor Rule for Semiconductor Manufacturing Processes". In: *2020 3rd International Conference on Intelligent Autonomous Systems (ICoIAS)*. IEEE. 2020, pp. 83–87.

[62] T. Nakazawa and D. V. Kulkarni. "Wafer map defect pattern classification and image retrieval using convolutional neural network". In: *IEEE Transactions on Semiconductor Manufacturing* 31.2 (2018), pp. 309–314.

[63] R. di Bella, D. Carrera, B. Rossi, P. Fragneto, and G. Boracchi. "Wafer Defect Map Classification Using Sparse Convolutional Networks". In: *International Conference on Image Analysis and Processing*. Springer. 2019, pp. 125–136.

[64] T. Tsigkritis, G. Groumas, and M. Schneider. "On the Use of k-NN in Anomaly Detection". In: *Journal of Information Security* (2018), pp. 635–645.

[65] L. C. Viagrande., F. L. M. Milotta., P. Giuffrè, G. Bruno., D. Vinciguerra., and G. Gallo. "Semisupervised Classification of Anomalies Signatures in Electrical Wafer Sorting (EWS) Maps". In: *Proceedings of the 15th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications - Volume 5: VISAPP,* INSTICC. SciTePress, 2020, pp. 278–285. ISBN: 978-989-758-402-2. DOI: 10.5220/0008914402780285.

[66] J. L. Leevy, T. M. Khoshgoftaar, R. A. Bauder, and N. Seliya. "A survey on addressing high-class imbalance in big data". In: *Journal of Big Data* 5.1 (2018), p. 42.

[67] C. Shorten and T. M. Khoshgoftaar. "A survey on image data augmentation for deep learning". In: *Journal of Big Data* 6.1 (2019), p. 60.

[68] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. "Gradient-based learning applied to document recognition". In: *Proceedings of the IEEE* 86.11 (1998), pp. 2278–2324.

[69] M. Shopon, N. Mohammed, and M. A. Abedin. "Image augmentation by blocky artifact in deep convolutional neural network for handwritten digit recognition". In: *2017 IEEE International Conference on Imaging, Vision & Pattern Recognition (icIVPR)*. IEEE. 2017, pp. 1–6.

[70] M. Frid-Adar, I. Diamant, E. Klang, M. Amitai, J. Goldberger, and H. Greenspan. "GAN-based synthetic medical image augmentation for increased CNN performance in liver lesion classification". In: *Neurocomputing* 321 (2018), pp. 321–331.

[71] X. Yin, Y. Li, X. Zhang, and B.-S. Shin. "Medical Image Augmentation Using Image Synthesis with Contextual Function". In: *2019 12th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*. IEEE. 2019, pp. 1–6.

[72] L. Perez and J. Wang. "The effectiveness of data augmentation in image classification using deep learning". In: *arXiv preprint arXiv:1712.04621* (2017).

[73] L. Taylor and G. Nitschke. "Improving deep learning using generic data augmentation". In: *arXiv preprint arXiv:1708.06020* (2017).

[74] H. Salehinejad, S. Valaee, T. Dowdell, and J. Barfett. "Image augmentation using radial transform for training deep neural networks". In: *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2018, pp. 3016–3020.

[75] H. Inoue. "Data augmentation by pairing samples for images classification". In: *arXiv preprint arXiv:1801.02929* (2018).

[76] R. Takahashi, T. Matsubara, and K. Uehara. "Data augmentation using random image cropping and patching for deep CNNs". In: *IEEE Transactions on Circuits and Systems for Video Technology* (2019).

[77] M. D. Bloice, C. Stocker, and A. Holzinger. "Augmentor: an image augmentation library for machine learning". In: *arXiv preprint arXiv:1708.04680* (2017).

[78] A. Buslaev, V. I. Iglovikov, E. Khvedchenya, A. Parinov, M. Druzhinin, and A. A. Kalinin. "Albumentations: fast and flexible image augmentations". In: *Information* 11.2 (2020), p. 125.

[79] C. Summers and M. J. Dinneen. "Improved mixed-example data augmentation". In: *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE. 2019, pp. 1262–1270.

[80] W. J. Frawley, G. Piatetsky-Shapiro, and C. J. Matheus. "Knowledge discovery in databases: An overview". In: *AI magazine* 13.3 (1992), pp. 57–57.

[81] J. S. Deogun, V. V. Raghavan, A. Sarkar, and H. Sever. "Data mining: Research trends, challenges, and applications". In: *Roughs Sets and Data Mining: Analysis of Imprecise Data* (1997), pp. 9–45.

[82] R. Agrawal, T. Imieliński, and A. Swami. "Mining association rules between sets of items in large databases". In: *Proceedings of the 1993 ACM SIGMOD international conference on Management of data*. 1993, pp. 207–216.

[83] A. Raorane, R. Kulkarni, and B. Jitkar. “Association rule–extracting knowledge using market basket analysis”. In: *Research Journal of Recent Sciences* 2277 (2012), p. 2502.

[84] J. Pillai and O. Vyas. “Overview of itemset utility mining and its applications”. In: *International Journal of Computer Applications* 5.11 (2010), pp. 9–13. DOI: 10.5120/956-1333.

[85] R. Agrawal, R. Srikant, et al. “Fast algorithms for mining association rules”. In: *Proc. 20th int. conf. very large data bases, VLDB*. Vol. 1215. 1994, pp. 487–499.

[86] M. Al-Maolegi and B. Arkok. “An improved apriori algorithm for association rules”. In: *arXiv preprint arXiv:1403.3948* (2014).

[87] S. Panjaitan, M. Amin, S. Lindawati, R. Watrianthos, H. T. Sihotang, B. Sinaga, et al. “Implementation of Apriori Algorithm for Analysis of Consumer Purchase Patterns”. In: *Journal of Physics: Conference Series*. Vol. 1255. 1. IOP Publishing. 2019, p. 012057.

[88] C. Borgelt. “An Implementation of the FP-growth Algorithm”. In: *Proceedings of the 1st international workshop on open source data mining: frequent pattern mining implementations*. 2005, pp. 1–5.

[89] R. Ranjan and A. Sharma. “Evaluation of frequent itemset mining platforms using apriori and fp-growth algorithm”. In: *International Journal of Information Systems & Management Science* 2.2 (2019).

[90] W. Kreesuradej and W. Thurachon. “Discovery of Incremental Association Rules Based on a New FP-Growth Algorithm”. In: *2019 IEEE 4th International Conference on Computer and Communication Systems (ICCCS)*. IEEE. 2019, pp. 184–188.

[91] J. Yabing. “Research of an improved apriori algorithm in data mining association rules”. In: *International Journal of Computer and Communication Engineering* 2.1 (2013), p. 25.

[92] H. Yu, J. Wen, H. Wang, and L. Jun. “An improved Apriori algorithm based on the Boolean matrix and Hadoop”. In: *Procedia Engineering* 15 (2011), pp. 1827–1831.

[93] X. Yuan. "An improved Apriori algorithm for mining association rules". In: *AIP conference proceedings*. Vol. 1820. 1. AIP Publishing LLC. 2017, p. 080005.

[94] S. A. Ahmed and B. Nath. "Modified FP-Growth: An Efficient Frequent Pattern Mining Approach from FP-Tree". In: *International Conference on Pattern Recognition and Machine Intelligence*. Springer. 2019, pp. 47–55.

[95] S. Ninoria and S. Thakur. "Review On Rare Itemset Mining". In: *International Journal of Computer Sciences and Engineering, NCRTI* (2019).

[96] R. Agrawal, T. Imielinski, and A. Swami. "Database mining: A performance perspective". In: *IEEE transactions on knowledge and data engineering* 5.6 (1993), pp. 914–925.

[97] B. Liu, W. Hsu, and Y. Ma. "Mining association rules with multiple minimum supports". In: *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*. 1999, pp. 337–341.

[98] J. C.-W. Lin, W. Gan, P. Fournier-Viger, T.-P. Hong, and V. S. Tseng. "Mining potential high-utility itemsets over uncertain databases". In: *Proceedings of the ASE BigData & SocialInformatics 2015*. 2015, pp. 1–6.

[99] J. Singh, H. Ram, and D. J. Sodhi. "Improving efficiency of apriori algorithm using transaction reduction". In: *International Journal of Scientific and Research Publications* 3.1 (2013), pp. 1–4.

[100] L. Szathmary, A. Napoli, and P. Valtchev. "Towards rare itemset mining". In: *19th IEEE International Conference on Tools with Artificial Intelligence (ICTAI 2007)*. Vol. 1. IEEE. 2007, pp. 305–312.

[101] H. Yun, D. Ha, B. Hwang, and K. H. Ryu. "Mining association rules on significant rare data using relative support". In: *Journal of Systems and Software* 67.3 (2003), pp. 181–191.

[102] C.-F. Chien, W.-C. Wang, and J.-C. Cheng. "Data mining for yield enhancement in semiconductor manufacturing and an empirical study". In: *Expert Systems with Applications* 33.1 (2007), pp. 192–198.

[103] A. Casali and C. Ernst. "Discovering correlated parameters in semiconductor manufacturing processes: A data mining approach". In: *IEEE Transactions on Semiconductor Manufacturing* 25.1 (2011), pp. 118–127.

[104] A. Kusiak. "Rough set theory: a data mining tool for semiconductor manufacturing". In: *IEEE transactions on electronics packaging manufacturing* 24.1 (2001), pp. 44–50.

[105] K. Nakata, R. Orihara, Y. Mizuoka, and K. Takagi. "A comprehensive big-data-based monitoring system for yield enhancement in semiconductor manufacturing". In: *IEEE Transactions on Semiconductor Manufacturing* 30.4 (2017), pp. 339–344.

[106] L. C. Viagrande, D. Allegra, and F. Stanco. "Curve Design Studio: Bézier Curve Integrated Tool for Video Game Development". In: *Smart Tools and Apps for Graphics - Eurographics Italian Chapter Conference*. Ed. by S. Biasotti, R. Pintus, and S. Berretti. The Eurographics Association, 2020. ISBN: 978-3-03868-124-3. DOI: 10.2312/stag.20201236.

[107] L. C. Viagrande, D. Allegra, and F. Stanco. "Point to Culture: a Point to Click Framework for Serious Games in Cultural Heritage". In: *Smart Tools and Apps for Graphics - Eurographics Italian Chapter Conference*. Ed. by S. Biasotti, R. Pintus, and S. Berretti. The Eurographics Association, 2020. ISBN: 978-3-03868-124-3. DOI: 10.2312/stag.20201248.

[108] Carter, Nathan. *Introduction to the Mathematics of Computer Graphics*. Vol. 51. American Mathematical Soc, 2016.

[109] Dobashi, Yoshinori. "Mathematics in Computer Graphics". In: *Mathematical Insights into Advanced Computer Graphics Techniques* (2019), 1.

[110] Han, JungHyun. *3D graphics for game programming*. Chapman and Hall/CRC, 2011.

[111] Raja, SP. "Bézier and B-Spline Curves - a Study and its application in Wavelet Decomposition". In: *International Journal of Wavelets, Multiresolution and Information Processing* (2020).

[112] Salomon, David. *Curves and surfaces for computer graphics*. Packt Publishing Ltd, 2006. DOI: {10.1007/0-387-28452-4}.

[113] Farin, Gerald E and Farin, Gerald. *Curves and surfaces for CAGD: a practical guide*. Morgan Kaufmann, 2002.

[114] Mortenson, Michael E. *Mathematics for computer graphics applications*. Industrial Press Inc., 1999.

[115] Scheiderer, Christian and Thun, Timo and Meisen, Tobias. "Bézier Curve Based Continuous and Smooth Motion Planning for Self-Learning Industrial Robots". In: *Procedia Manufacturing* 38 (2019), 423–430.

[116] Erleben, Kenny and Henriksen, Knud. "Scripted bodies and spline driven animation". In: *Graphics programming methods* (2002), 37–50.

[117] Khan, Murtaza Ali and Sarfraz, Muhammad. "Motion tweening for skeletal animation by cardinal spline". In: *International Conference on Informatics Engineering and Information Science*. Springer. 2011, 179–188.

[118] Berger, Matthias and Cristie, Verina. "CFD Post-processing in Unity3D". In: *Procedia Computer Science* 51 (2015), 2913–2922.

[119] WU, Xiao-liang and HUANG, Xiang-nian. "Using Bézier Curve to Bend 3D Objects in Unity". In: *Modern Computer* 7 (2016), 15.

[120] Hawley, Richard A. *Grome Terrain Modeling with Ogre3D, UDK, and Unity3D*. Packt Publishing Ltd, 2013.

[121] Chadimová, Lenka. "The creating serious games for historical subjects at the 1st level of primary school and such as a part of exhibition chosen historical buildings". In: *Procedia - Social and Behavioral Sciences* (2015).

[122] DIAZ, LIZZIE EDMEA NARVAEZ and PECH, VICTOR MANUEL CHI and CASTRO, ERIKA ROSSANA LLANES and EUAN, MAXIMILIANO CANCHE. "Address classifier: a game-based educational application for computer networks". In: *International Journal of Advanced Research in Computer Science and Software ...* (2018).

[123] De Carli, Daniel Michelon and Bevilacqua, Fernando and Pozzer, Cesar Tadeu and d'Ornellas, Marcos Cordeiro. "A survey of procedural content generation techniques suitable to game development". In: *2011 Brazilian Symposium on Games and Digital Entertainment*. IEEE. 2011, 26–35.

[124] de Toni, Rodrigo and Marson, Fernando and Cassol, Vinicius J. "Dynamic terrains applying pseudo-infinity and synthesized by Bézier curves". In: *Simp/'osio Brasileiro de Jogos e Entretenimento Digital*. 2016.

[125] Desmond, Michael and Bahana, Raymond. "Bézier VFX plug-in: Improving unity visual effect performance". In: *2016 1st International Conference on Game, Game Art, and Gamification (ICGGAG)*. IEEE. 2016, 1–6.

[126] Arkham Interactive. *Bézier Curve Editor*. https://assetstore.unity.com/packages/tools/bezier-curve-editor-11278.

[127] BansheeGz. *Bg Curve*. https://assetstore.unity.com/packages/tools/utilities/bg-curve-59043.

[128] Alexander Kotov. *Bézier Master*. https://assetstore.unity.com/packages/tools/level-design/bezier-master-107374.

[129] Benoit Dumas. *Spline Mesh*. https://assetstore.unity.com/packages/tools/modeling/splinemesh-104989.

[130] J. Xie. "Research on key technologies base Unity3D game engine". In: *2012 7th international conference on computer science & education (ICCSE)*. IEEE. 2012, pp. 695–699.

[131] Lorentz, George G. *Bernstein polynomials*. American Mathematical Soc., 2013.

[132] Prautzsch, Hartmut and Boehm, Wolfgang and Paluszny, Marco. *Bézier and B-spline techniques*. Springer Science & Business Media, 2013.

[133] Rockwood, Alyn and Chambers, Peter. "Interactive curves and surfaces". In: *Technology-Based Re-Engineering Engineering Education Proceedings of Frontiers in Education FIE'96 26th Annual Conference*. Vol. 1. IEEE. 1996, 471–474.

[134] Khan, Khalid and Lobiyal, DK and Kilicman, Adem. "A de Casteljau Algorithm for Bernstein type Polynomials based on (p, q)-integers." In: *Applications & Applied Mathematics* 13.2 (2018).

[135] S. Bampatzia, I. Bourlakos, A. Antoniou, C. Vassilakis, G. Lepouras, and M. Wallace. "Serious games: valuable tools for cultural heritage". In: *International Conference on Games and Learning Alliance*. Springer, 2016, pp. 331–341.

[136] B. Bontchev et al. "Serious games for and as cultural heritage". In: *Digital Presentation and Preservation of Cultural and Scientific Heritage* (2015), pp. 43–58.

[137] T. Coenen, L. Mostmans, and K. Naessens. "MuseUs: Case study of a pervasive cultural heritage serious game". In: *Journal on Computing and Cultural Heritage (JOCCH)* 6.2 (2013), pp. 1–19.

[138] T. Ekonomou and S. Vosinakis. "Mobile augmented reality games as an engaging tool for cultural heritage dissemination: A case study". In: *Sci. Cult* 4 (2018), pp. 97–107.

[139] M. Mortara, C. E. Catalano, F. Bellotti, G. Fiucci, M. Houry-Panchetti, and P. Petridis. "Learning cultural heritage by serious games". In: *Journal of Cultural Heritage* 15.3 (2014), pp. 318–325.

[140] C. Tsita and M. Satratzemi. "How serious games in cultural heritage are being evaluated". In: *Proceedings of the 11th Pan-Hellenic & International Conference, ICT in Education (HCICTE), Thessaloniki*. 2018.

[141] J. I. Hittorff and L. Zath. *Architecture moderne de la Sicile, ou Recueil des plus beaux monumens religieux et des édifices publics et particuliers les plus remarquables de la Sicile*. 1835.

[142] D. Allegra, G. Gallo, L. Inzerillo, M. Lombardo, F. L. M. Milotta, C. Santagati, and F. Stanco. "Low Cost Handheld 3D Scanning for Architectural Elements Acquisition". In: *Smart Tools and Apps for Graphics*. 2016. DOI: 10.2312/stag.20161372.

[143] A. Angelopoulou, D. Economou, V. Bouki, A. Psarrou, L. Jin, C. Pritchard, and F. Kolyda. "Mobile Augmented Reality for Cultural Heritage". In: *Mobile Wireless Middleware, Operating Systems, and Applications*. 2012, pp. 15–22.

[144] R. Garozzo, G. Pasqualino, D. Allegra, C. Santagati, and F. Stanco. "Augmented Reality for the Valorization and Communication of Ruined Architecture". In: *International Conference on Image Analysis and Processing*. 2019, pp. 170–178. DOI: 10.1007/978-3-030-30754-7_17.

[145] I. Malegiannaki and T. Daradoumis. “Analyzing the educational design, use and effect of spatial games for cultural heritage: A literature review”. In: *Computers & education* 108 (2017), pp. 1–10.

[146] R. Andreoli, A. Corolla, A. Faggiano, D. Malandrino, D. Pirozzi, M. Ranaldi, G. Santangelo, and V. Scarano. “A framework to design, develop, and evaluate immersive and collaborative serious games in cultural heritage”. In: *Journal on Computing and Cultural Heritage (JOCCH)* 11.1 (2017), pp. 1–22.

[147] A. Antoniou, G. Lepouras, S. Bampatzia, and H. Almpanoudi. “An approach for serious game development for cultural heritage: Case study for an archaeological site and museum”. In: *Journal on Computing and Cultural Heritage (JOCCH)* 6.4 (2013), pp. 1–19.

[148] F. Bellotti, R. Berta, A. De Gloria, A. D’ursi, and V. Fiore. “A serious game model for cultural heritage”. In: *Journal on Computing and Cultural Heritage (JOCCH)* 5.4 (2013), pp. 1–27.

[149] S. Moya, D. Tost, S. Grau, A. von Barnekow, and E. Felix. “SKETCH’NDO: A framework for the creation of task-based serious games”. In: *Journal of Visual Languages & Computing* 34 (2016), pp. 1–10.

[150] J. Suominen and A. Sivula. “Gaming legacy? four approaches to the relation between cultural heritage and digital technology”. In: *Journal on Computing and Cultural Heritage (JOCCH)* 6.3 (2013), pp. 1–18.

[151] M. Vayanou, Y. Ioannidis, G. Loumos, and A. Kargas. “How to play storytelling games with masterpieces: From art galleries to hybrid board games”. In: *Journal of Computers in Education* 6.1 (2019), pp. 79–116.

[152] I. Paliokas and S. Sylaiou. “A Classification Model for Serious Games Used in Museums, Galleries and Other Cultural Sites”. In: *Strategic Innovative Marketing and Tourism*. Springer, 2019, pp. 1057–1064.

[153] S. Rizvic, D. Boskovic, V. Okanovic, S. Sljivo, and M. Zukic. “Interactive digital storytelling: bringing cultural heritage in a classroom”. In: *Journal of Computers in Education* 6.1 (2019), pp. 143–166.

[154] S. Pavkov, I. Franković, and N. Hoić-Božić. "Comparison of game engines for serious games". In: *2017 40th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*. IEEE. 2017, pp. 728–733.

[155] J. Tomalá-Gonzáles, J. Guamán-Quinche, E. Guamán-Quinche, W. Chamba-Zaragocin, and S. Mendoza-Betancourt. "Serious Games: Review of methodologies and Games engines for their development". In: *2020 15th Iberian Conference on Information Systems and Technologies (CISTI)*. IEEE. 2020, pp. 1–6.

[156] I. Studios. *Adventure Creator*. https://assetstore.unity.com/packages/templates/systems/adventure-creator-11896.

[157] K. C. Bul, I. H. Franken, S. Van der Oord, P. M. Kato, M. Danckaerts, L. J. Vreeke, A. Willems, H. J. Van Oers, R. Van den Heuvel, R. Van Slagmaat, et al. "Development and user satisfaction of "Plan-It Commander," a serious game for children with ADHD". In: *Games for health journal* 4.6 (2015), pp. 502–512.