Indonesian Journal of Electrical Engineering and Computer Science
Vol. 26, No. 1, April 2022, pp. 143~151
ISSN: 2502-4752, DOI: 10.11591/ijeecs.v26.i1.pp143-151

Electronic datasheet parameter extractor and verifier system for printed circuit board development

Ma. Chalina S. Cntapay, Mark Joseph B. Enojas, Hohn Lois C. Bongao
Bachelor of Engineering and Allied Department, Technological University of the Philippines, Taguig, Philippines

Article Info

Article history:

Received Dec 2, 2021
Revised Feb 8, 2022
Accepted Feb 12, 2022

Keywords:

Component library
Datasheet
PCB footprint
Pin configuration
Text recognition

ABSTRACT

Library components are composed of devices, symbols, and footprints which are the fundamental parts to design and develop printed circuit boards (PCB). In archiving PCB libraries, the creation and checking of the component's parameters are very important. The checking of footprints and pin configurations from datasheets are manually done which causes delays and downtime when the components do not fit the developed PCB. It was found in a survey that 61.5% of errors come from footprints dimensions. These cause delays in the production and development of PCBs. In this study, a parameter extractor and verifier system were developed to mitigate the error of manual process of archiving library parameters from datasheets. Text recognition and pattern matching algorithms were used for the processing of images from datasheets. The 10-trial tests conducted for each footprint for the extraction of text from the cropped images for standard operating procedures (SOP) series and heat sink thin shrink small outline package (HTSSOP) series were found successful without any error. The checking of footprints and pin configuration was reduced from 45 minutes to 25 minutes. The developed system was evaluated based on user perception by 10 PCB library users which resulted to agree that the system is functional, efficient, and convenient to use.

Corresponding Author:

Ma. Chalina S. Cntapay
Bachelor of Engineering and Allied Department, Technological University of the Philippines
Taguig Philippines
Email: machalina.cuntapay@tup.edu.ph

1. INTRODUCTION

The evolution of the design and development of printed circuit boards (PCBs) emerges with the changes in integrated circuit (IC) packages and other electronic devices [1]. The footprint and pin configuration on PCB designs from pin-holed components change as surface-mount and lightweight components evolve. Manufacturers have developed library systems for easy configuration of devices from those available locally and online. To reuse components, library systems are developed [2]. This is also applied to electronic components archiving and development of PCBs [3], [4].

The electronic circuits' PCB design undergoes three stages namely, libraries, schematics, and board. The fundamental part in PCB design is the library. In the library can be found devices, symbols, and footprints of electronic components such as ICs [5], [6]. The components' footprints and symbols are designed with schematics. The footprints are the arrangement of pads that are physically and electrically connected which are composed of parameters and details that help the designer to identify easily which parts need to be modified. The board plays an important role where the IC footprints [7] must be fitted well. Even in the simulation for the development of micro-electro-mechanical (MEMS), component libraries must be present [8]. The footprint components' information must be in place even when designs vary depending on

the manufacturer [9]. This information, including the manufacturer's name, can be included as parameters in the datasheets for package classification [10].

Standardization of electronic component footprints and other features must be established in order to allow designers optimize their PCB designs and schematics [11]. However, the uniqueness of components being developed may vary because of diverse applications. Datasheets and drawings come in different formats from different electronic components manufacturers. In order for these products to be archived in a library, the user needs to input it manually by reading the data sheet and encoding all the parameter values of the foot print and pin configuration. Human error always exists in manual processes like this.

A survey was conducted in one of the leading electronic components manufacturers in the Philippines which shows that 61.5% of the PCB designers and librarians agreed that oftentimes, the encoding of the dimensions of the footprints are the common sources of errors. Dimensions are critical in the design of PCBs because the copper track layout and spacing between pads rely on it [12]. The inspection of PCBs relies on the data archived in the library. It is time consuming and considered a downtime when the parameters did not match with the PCB designed and developed which can turn out into waste [13]–[15]. Visual component libraries play a vital role for optimized designs minimizing errors and cost in PCB development [16], [17].

This paper presents a method of automating the data parameters extraction and verification from the data sheet drawings for archiving it in a library of electronic components. An image processing, text recognition, and pattern matching algorithm were used to extract the parameters from an automatically cropped image from data sheet drawings. In this way, the error and time consumed in checking footprints and pin configuration will be lessened. The developed system is a web pack running in a local network.

2. METHODOLOGY

2.1. The IC PCB footprint and pin configuration parameter extractor and verifier system (PEVS)

The process of the parameter extractor and verifying system is presented in Figure 1, which has five parts: the load image (LD) for image acquisition; extraction using the application protocol interface (API) commands; classification; optimization; printing and displaying. The extraction undergoes three subprocesses: the preprocessing for filtration and binarization of image, the segmentation for character recognition, and the feature extraction. The image post processing output will be fed to the library or can be a printable and readable display.

Text recognition technology has been widely used since it was discovered. Optical character recognition (OCR) has been extensively developed using different technologies [18]–[23] even for much more advanced techniques [24]–[26]. In this system, tesseract, which is an open-source OCR engine, is used because of its excellent capability of recognizing six western languages. TesseractJS is a plugin integrated in the system which can run in a browser or server with NodeJs.

The PEVS has three display windows: the main window, the middle, and the training window. In the main window, the user can view the result of the two images being compared, where the image post-processing output is displayed. In the middle window is where the training datasets or images can be viewed. In the training window is where the training is done by extracting the parameters from the image. It is where the pdf files are loaded, cropped, and converted into an image file.

2.2. Training and verification setup

It is imperative that the character recognition must be set up well. VUE.js, which has an open-source software is used for building user-machine interface. Within a single page, these interfaces can be viewed. By building this framework, the PDF file will be loaded to the system for parameters training. It will be cropped and converted into an image file format. The pattern matching algorithm is used to look for a reference in the drawing to identify points in the edges for cropping. The region of interest which are the blue rectangles will be set up in place for character recognition.

Tesseract is used as a plugin for text recognition and image processing. It also has a classification system in which the images are compared with the pre-defined patterns for parameter detection. Once done, the trained setup will be saved. See Figure 2 for the pseudo code. The original and the process image are being displayed in a viewport for verification. The platform used as a front end is made using hypertext processor (PHP) along with vue.js for web development in a local network. The results will be shown in percent similarity.

2.3. Time Study

Automating the processes can also be evaluated through time studies, where the time spent for the automated process is being compared to the previous process. The time spent for processing one data sheet

for parameter extraction and verification for both footprint and pin configuration were recorded and compared. The results must verify that the downtime is reduced.

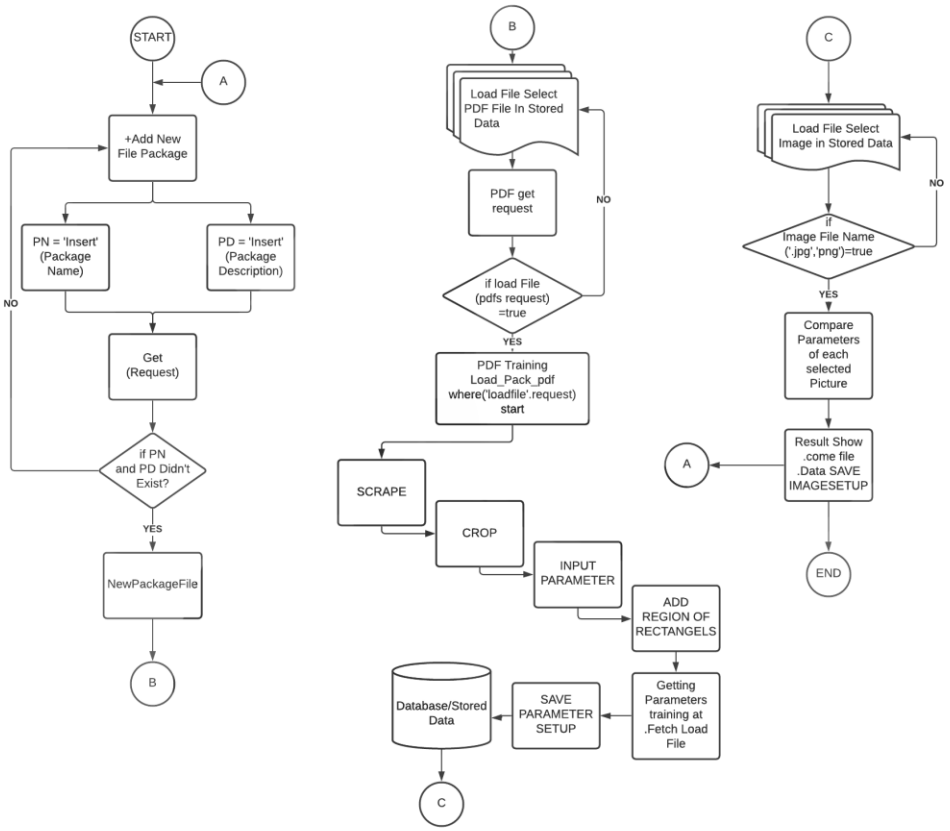


Figure 1. Process flow chart of PEVS

```
Pseudo Code for Datasheet Parameter Extractor System

1.    Initialize package PN = package name and  PD= package description
2.    Package = (Set Initial Package Value:);
3.    Display Responses;
4.    "Package_Name:"
5.    "Package Description:"
6.        {
7.         Case 1: //For Package name
8.              IF Package name = 1 (true), proceed
9.                 (Package Name Exist);
10             ELSE package name = 0 (false), proceed
11.               Set the Package name edit the value
12.       Case 2: //For Package description
13.             IF Package description = 1 (true), proceed
14.                (Package description);
15.             ELSE Package description = 0 (false), proceed
16.                Set the Package description edit the value
17.       }
18.   Choose and Load PDF file = Data storage
19.   Select available PDF files, process
20.   Initialize PDF file
21.      {
22.       IF
23.             Load PDF file isn't recognized go to step 18 to Choose file
24.             print response, "The file you choose doesn't response"
25.       ELSE
26.             PDF training, proceed
27.             scrape, getting position to crop
28.             Print "Display image output"
29.      }
30.   Input Parameters one at a time
31.   Add region of rectangles at the Display Image
32.   Training of parameters done
33.   Save Training ( to Data stored)
34.   Select images from training data stored, decision
35.   Print response "Display responses of images"
36.   "Image 1" = "Created footprint"
37.   "Image 2" = "Datasheet"
38.   Comparing of 2 images and its parameters
39.   Print response "Result Shown"
40.   All data save at Data stored
```

Figure 2. Pseudo code of the developed PEVS

3. RESULTS AND DISCUSSION

3.1. PVES development

The developed system is deployed in a local network and has a package display window as presented in Figure 3. The package name and description of the specific IC or electronic component will be needed. For testing purposes, three types of library components were used, namely the heat sink thin shrink small outline package (HTSSOP) series, standard operating procedures (SOP) series, and VQFP48C series. These are integrated circuits which have different pin configurations and footprints.

Once the package is set, it will be loaded and cropped for processing the image. The pattern matching algorithm will identify the position of the drawing and crop it accordingly with an identified size. When cropped, the rectangles are positioned as regions of interest for character recognition. It will be converted into an image file and saved in a database for archiving. The cropping and converting images are presented in Figure 4. The datasheet is being saved first, as presented in Figure 4(a). In Figure 4(b), the pdf file was loaded. It undergoes pattern matching for cropping into an image file which in result is now ready for text recognition as presented in Figure 4(c).

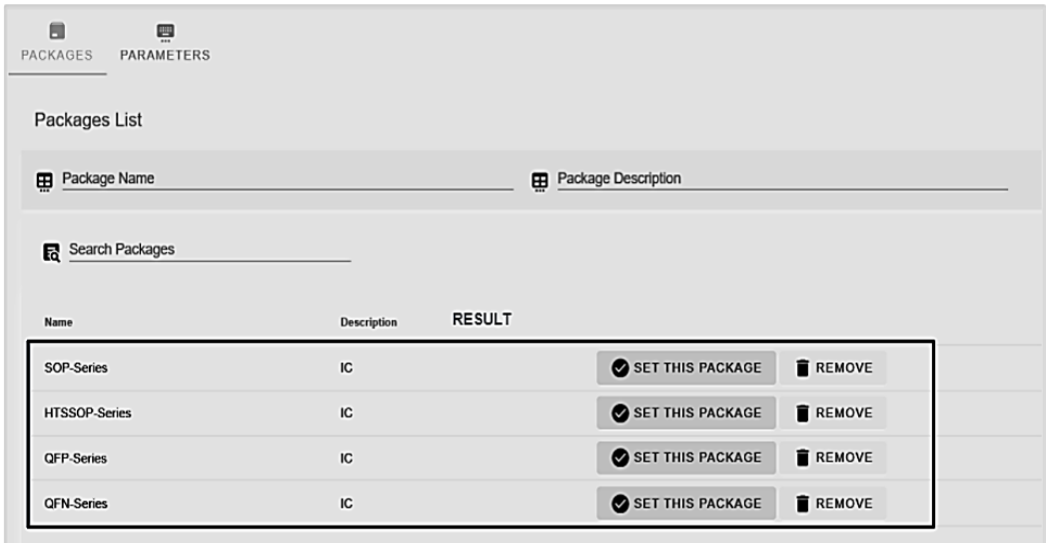
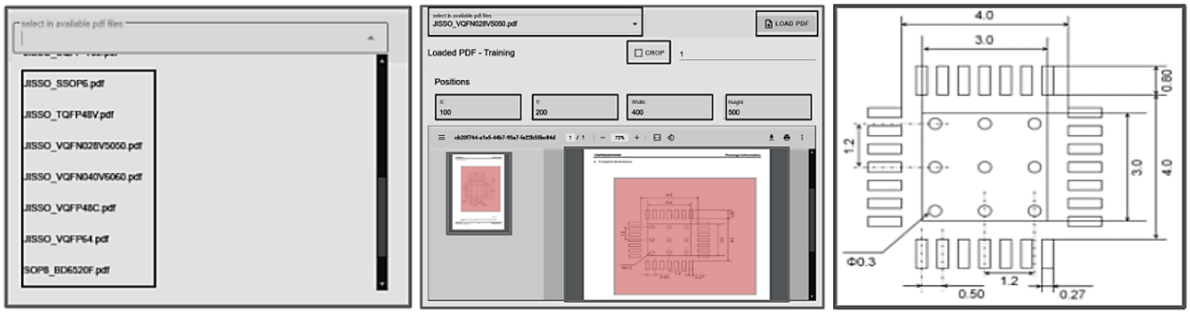


Figure 3. The package display window



(a) (b) (c)

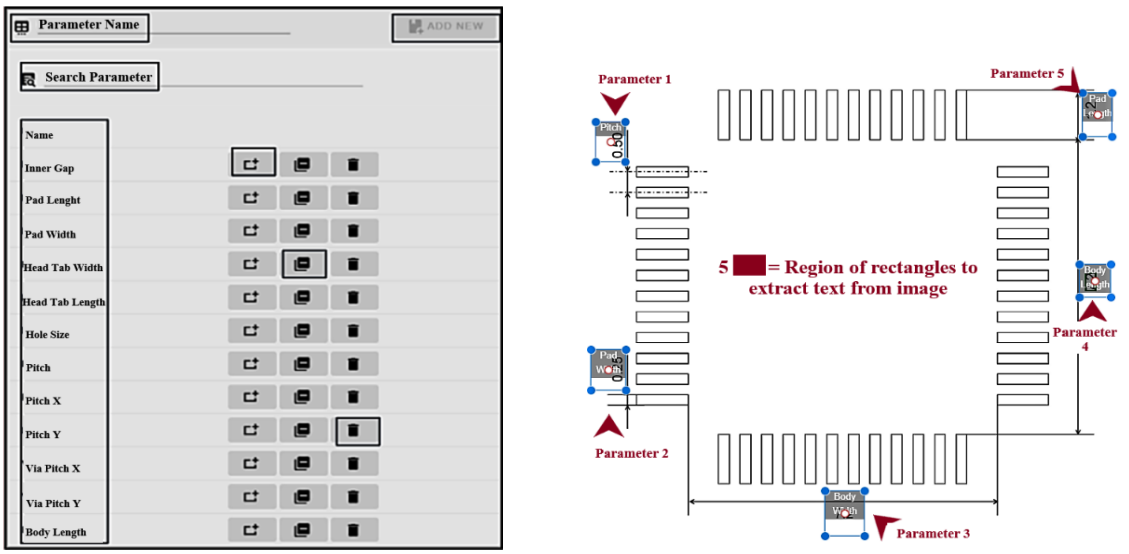
Figure 4. Cropping and converting of pdf file; (a) datasheet saving, (b) loading and cropping, and (c) converted pdf zinto an image file

3.2. Parameter extraction

The extraction of parameter values from the cropped image is done by putting rectangular boxes as regions of interest for text recognition. During the training stage, the parameters are written manually and the rectangular boxes are positioned manually as well. When successful, the system will automatically analyze the data sheet fed for analysis.

The parameter names of the components are put in the command box. All the parameters set by the operator will be added to the classification system as presented in Figure 5. Figure 5(a) presents an example of the extraction results. In this case, there are 10 parameters listed and saved for archiving. Each designated rectangle on the cropped image has designated parameters, such as inner gap, pad length, pad width, heat tab width, heat tab length, pitch, via pitch, hole size, and body length. For electronic packages with 5 parameters, other parameters can be simply added or deleted in the command box, as presented in Figure 5(b).

As an example, the dimension parameters are user-defined circuit dimensions, which are critically identified because it allows the analysis engine to modify the circuit to perform parameter checks and optimization. It provides a faster way for the designer to change dimensions in the project editor, PCB Cadence, as shown in Figure 6. The parameters identified will also be used for verification.



(a) (b)

Figure 5. Parameter extraction (a) parameter command window and (b) 5-parameter settings with assigned region of interest

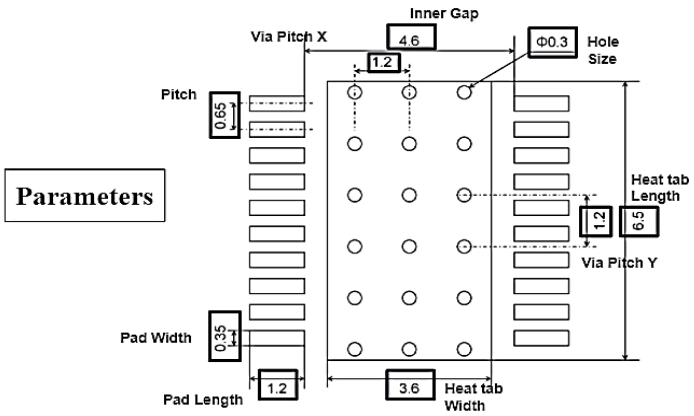


Figure 6. Parameters in a footprint by PCB cadence

3.3. Verification

The verification system has the following functions: compare, set, and search functions. The preliminary test result for footprint parameters comparison between SOP8 products from different manufacturers matches identically as seen in Figure 7. The image comparison for the same product from different manufacturers is 71.43% simular. The verification of pin configuration for two datasheets of SOP8

from different manufacturers are presented in Figure 8. The two images are 71.88% similar. The texts were successfully recognized for the two images.

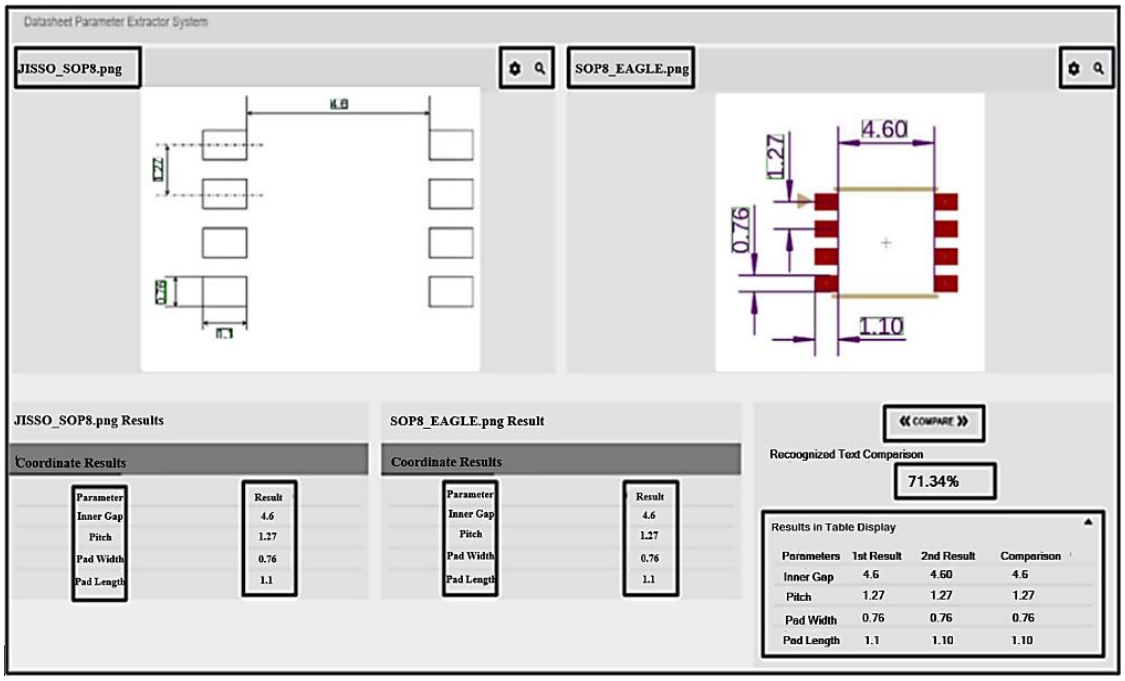


Figure 7. Main window with the package parameters and preliminary results for footprint comparison

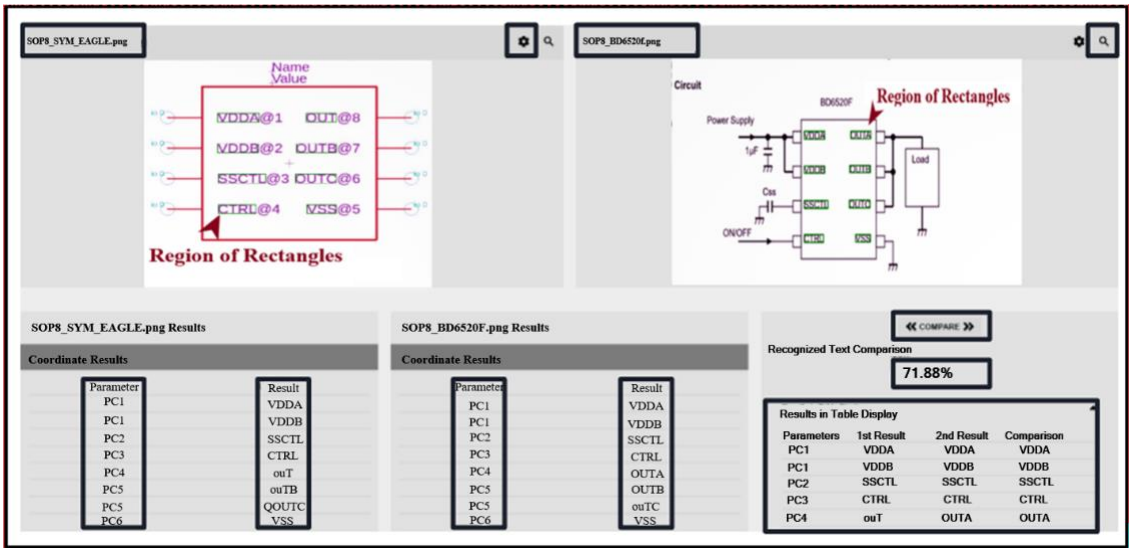


Figure 8. Main window with the package parameters and preliminary results for pin configuration comparison

3.4. Clicks and time study analysis of manual vs. PEVS checking of footprint and pin configuration

The number of clicks for checking footprints and pin configuration between manual and PEVS are recorded for comparison. The process of checking components by listing numbers of pins is recorded. The data are taken from the average of 10 users. The average time spent for checking IC footprints is 45 minutes as tabulated in Table 1. The manual checking of HTSSOP-B20 footprint, a 20-pin IC with 9 parameters is 30 clicks. The designer spends 20.4 minutes checking it. Using the developed PEVS, an average of 22 clicks or 10.75 minutes for the HTSSOP-B20. A total of 58 clicks were observed for checking the footprint of

HTSSOP-B20, SOP8, and VQFP48C using the PEVS. It is faster compared to that of the manual process which has a total of 72 clicks. The pin configuration checking using the manual method averages to 45 clicks in 45 minutes. Using the PEVS, only 35 clicks on average are made for 28 minutes. See Table 2 for the pin configuration checking results. A total of 120 clicks were observed using the PEVS in checking the pin configuration of the three components which is way faster than the manual checker with 175 clicks.

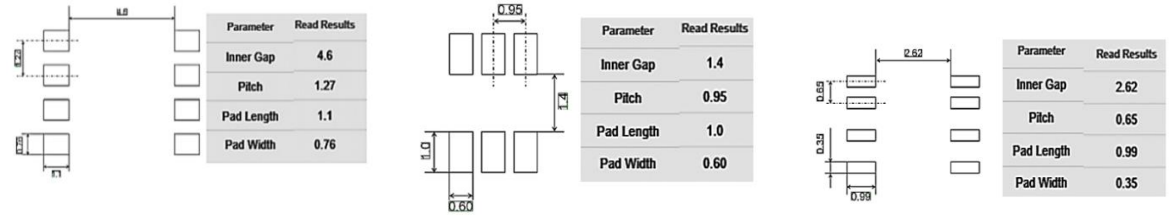
Table 1. Datasheet footprint checking and verificatation

Library Component	Manual Checking No. of Clicks	PEVS No. of Clicks
HTSSOP-B20	30	22
SOP8	20	16
VQFP48C	48	20
Total	72	58

Table 2. Datasheet pin configuration checking and verificatation

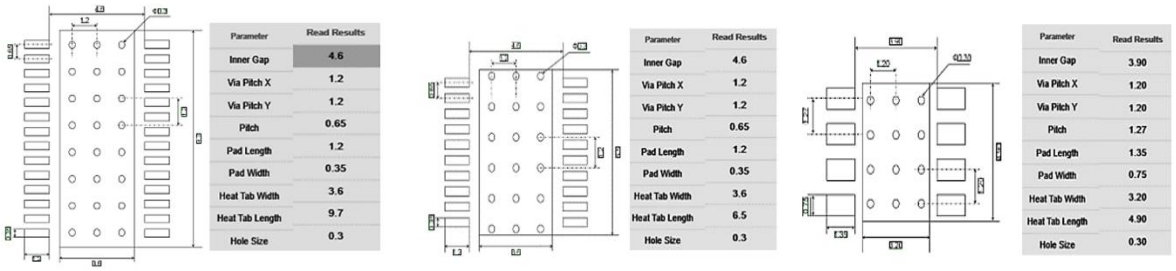
Library Component	Manual Checking No. of Clicks	PEVS No. of Clicks
HTSSOP-B20	45	35
SOP8	25	20
VQFP48C	105	65
Total	175	120

Additional tests were conducted for a series of library components for footprint checking. The results of checking of SOP-series ICs are presented in Figure 9 as follows: the SSOP6 in Figure 9(a), MSOP8 in Figure 9(b), and SOP8 in Figure 9(c). There were 10 trials conducted for each of these components. Same goes with the HTSSOP-series in Figure 10, where HTSSOP-B28 in Figure 10(a), HTSSOP-B20 in Figure 10(b), and HTSSOP-J8 in Figure 10(c) were tested. The results were summarized in Table 3. The 3 preliminary tests were conducted for training. The remaining tests determine the efficiency of the developed system. As a result, the SOP8 and HTSSOP-B20 fail in the first two tests. The HTSSOP-J8 also failed in the first test while HTSSOP-B28 fails until the third test. The succeeding test passed in which the system was verified to be working properly as expected.



(a) (b) (c)

Figure 9. SOP-series footprint checking (a) MSOP8, (b) SSOP6, and (c) SOP8



(a) (b) (c)

Figure 10. HTSSOP-series footprint checking (a) HTSSOP-B28, (b) HTSSOP-B20, and (c) HTSSOP-J8

Table 3. Additional test for footprint checking for SOP and HTSSOP series

Library Component	1	2	3	4	5	6	7	8	9	10
SOP8	F	F	P	P	P	P	P	P	P	P
MSOP8	P	P	P	P	P	P	P	P	P	P
SSOP6	P	P	P	P	P	P	P	P	P	P
HTSSOP-J8	F	P	P	P	P	P	P	P	P	P
HTSSOP-B20	F	F	P	P	P	P	P	P	P	P
HTSSOP-B28	F	F	F	P	P	P	P	P	P	P

4. CONCLUSION

This study developed a PEVS which is able to improve the process of parameter extraction and verification for archiving electronic component footprint and pin configuration in a library. The average of the manual process which is 45 minutes for IC footprint checking was improved to 10.75 minutes, while the pin configuration checking and verification was improved from an average of 45 minutes to 28 minutes. Using the technique of image post-processing by pattern matching and text recognition, the parameters can now be generated from the pdf file types of datasheets of electronic components. Based on the conducted tests, the system was able to extract parameters from different components, the SOP8 series and the HTSSOP series. The inputting of parameters to the library system had mitigated errors due to the manual process. The PEVS was deployed in a local network so that the librarians can easily access it. The end-to-end integration to a library software will be done in the future to extend the application of the developed system and will be deployed online.

ACKNOWLEDGEMENTS

The authors acknowledge the help of the Lord Jesus Christ and the faculty of the Bachelor of Engineering and Allied Department of the Technological University of the Philippines Taguig.

REFERENCES

[1] J. Kloeser and W. Scheel, "PCB-design follows IC-packaging," *Circuit World*, vol. 26, no. 3, pp. 6–10, Sep. 2000, doi: 10.1108/03056120010322834.

[2] H. M. Zhang and M. Guo, "A component library information model supporting component composition," in *2012 IEEE International Conference on Mechatronics and Automation, ICMA 2012*, Aug. 2012, pp. 475–479, doi: 10.1109/ICMA.2012.6282956.

[3] Y. Ni, Y. Wang, and T.-Y. Ho, "Footprint Classification of Electric Components on Printed Circuit Boards," in *2020 ACM/IEEE 2nd Workshop on Machine Learning for CAD (MLCAD)*, pp. 1–6, 2020.

[4] V. W. Ng, M. D. Seeman, and S. R. Sanders, "Minimum PCB footprint point-of-load DC-DC converter realized with Switched-Capacitor architecture," in *2009 IEEE Energy Conversion Congress and Exposition, ECCE 2009*, Sep. 2009, pp. 1575–1581, doi: 10.1109/ECCE.2009.5316488.

[5] D. Zhang, S. Zhang, and Y. Su, "The foundation of microcircuits DPA photo library and the contribution to the electronic product PHM," in *Proceedings of 2014 Prognostics and System Health Management Conference, PHM 2014*, Aug. 2014, pp. 669–672, doi: 10.1109/PHM.2014.6988257.

[6] S. U. Rehman, K. F. Thang, and N. S. Lai, "Automated PCB identification and defect-detection system (APIDS)," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 9, no. 1, pp. 297–306, 2019, doi: 10.11591/ijece.v9i1.pp297-306.

[7] J. Ou, A. Maldonado, C. Saephan, F. Farahmand, and M. Caggiano, "A low-cost PCB fabrication process," in *Proceedings - Electronic Components and Technology Conference*, May 2014, pp. 2159–2162, doi: 10.1109/ECTC.2014.6897601.

[8] W. Hu, X. Wei, and X. Xie, "Development of MEMS component library with modelica," in *ICCASM 2010 - 2010 International Conference on Computer Application and System Modeling, Proceedings*, Oct. 2010, vol. 2, doi: 10.1109/ICCASM.2010.5620369.

[9] P. T. Huang, X. Y. Lin, Y. J. Wang, and T. Y. Ho, "Ensemble Learning Based Electric Components Footprint Analysis," Aug. 2021, doi: 10.1109/MLCAD52597.2021.9531219.

[10] Y. J. Wang *et al.*, "An Artificial Neural Network to Support Package Classification for SMT Components," in *2018 3rd International Conference on Computer and Communication Systems, ICCCS 2018*, Apr. 2018, pp. 173–177, doi: 10.1109/CCOMS.2018.8463252.

[11] P. Peterson and J. Placente, "Component standardization in a distributed workstation CAD environment," in *Proceedings of the IEEE/CPMT International Electronics Manufacturing Technology (IEMT) Symposium*, 1994, vol. 1, pp. 103–107, doi: 10.1109/iemt.1994.404683.

[12] T. S. Kukal, A. Mathur, J. K. Ahuja, and S. Mohan, "Electrical overstress estimation for printed circuit board design," in *Proceedings-Annual Reliability and Maintainability Symposium*, Jan. 2020, vol. 2020-Janua, doi: 10.1109/RAMS48030.2020.9153638.

[13] I. A. Soomro, A. Ahmad, and R. H. Raza, "Printed Circuit Board identification using Deep Convolutional Neural Networks to facilitate recycling," *Resources, Conservation and Recycling*, vol. 177, p. 105963, Feb. 2022, doi: 10.1016/j.resconrec.2021.105963.

[14] F. J. Langley, R. R. Boatright, and L. Crosby, "Composite electro-optical testing of surface-mount device boards-one manufacturer's experience," in *International Test Conference 1989*, 1989, pp. 686–691.

[15] V. Raghuvanshi, A. Burman, P. P. Bartakke, and S. Deshpande, "PCB solder pad inspection mechanism using gerber file," in *International Conference on Communication and Signal Processing, ICCSP 2016*, Apr. 2016, pp. 1321–1325, doi: 10.1109/ICCSP.2016.7754367.

[16] O. L. Wang and J. Huang, "Developing a visual component library for a graphical programming platform using object orientation," *IEEE Aerospace and Electronic Systems Magazine*, vol. 17, no. 6, pp. 16–19, Jun. 2002, doi: 10.1109/MAES.2002.1010116.
[17] "Reducing PCB design costs: From schematic capture to PCB layout," 2005, Accessed: Feb. 21, 2022. [Online]. Available: www.ti.com/automotive.
[18] R. Parthiban, R. Ezhilarasi, and D. Saravanan, "Optical Character Recognition for English Handwritten Text Using Recurrent Neural Network," *2020 International Conference on System, Computation, Automation and Networking (ICSCAN)*, Jul. 2020, doi: 10.1109/ICSCAN49426.2020.9262379.
[19] T. K. Hazra, D. P. Singh, and N. Daga, "Optical character recognition using KNN on custom image dataset," in *2017 8th Industrial Automation and Electromechanical Engineering Conference, IEMECON 2017*, Aug. 2017, pp. 110–114, doi: 10.1109/IEMECON.2017.8079572.
[20] A. M. Sabu and A. S. Das, "A Survey on various Optical Character Recognition Techniques," in *Proc. IEEE Conference on Emerging Devices and Smart Systems, ICEDSS 2018*, Mar. 2018, pp. 152–155, doi: 10.1109/ICEDSS.2018.8544323.
[21] P. A. Khaustov, V. G. Spitsyn, and E. I. Maksimova, "Algorithm for optical handwritten characters recognition based on structural components extraction," in *Proceedings - 2016 11th International Forum on Strategic Technology, IFOST 2016*, Jun. 2017, pp. 355–358, doi: 10.1109/IFOST.2016.7884126.
[22] Y. M. Su, H. W. Peng, K. W. Huang, and C. S. Yang, "Image processing technology for text recognition," *2019 International Conference on Technologies and Applications of Artificial Intelligence (TAAI)*, Nov. 2019, doi: 10.1109/TAAI48200.2019.8959877.
[23] S. Shetty, A. S. Devadiga, S. Sibi Chakkaravarthy, and K. A. Varun Kumar, "Ote-OCR based text recognition and extraction from video frames," in *2014 IEEE 8th International Conference on Intelligent Systems and Control: Green Challenges and Smart Solutions, ISCO 2014 - Proceedings*, Jan. 2014, pp. 229–232, doi: 10.1109/ISCO.2014.7103949.
[24] D. Karthikeyan *et al.*, "Sophisticated and modernized library running system with OCR algorithm using IoT," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 24, no. 3, pp. 1680–1691, Dec. 2021, doi: 10.11591/ijeecs.v24.i3.pp1680-1691.
[25] O. M. Khanday and S. Dadvandipour, "Analysis of machine learning algorithms for character recognition: a case study on handwritten digit recognition," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 21, no. 1, pp. 574–581, Jan. 2021, doi: 10.11591/ijeecs.v21.i1.pp574-581.
[26] C. L. Liu, K. Nakashima, H. Sako, and H. Fujisawa, "Handwritten digit recognition: Benchmarking of state-of-the-art techniques," *Pattern Recognition*, vol. 36, no. 10, pp. 2271–2285, Oct. 2003, doi: 10.1016/S0031-3203(03)00085-2.

BIOGRAPHIES OF AUTHORS

Ma. Chalina S. Cuntapay received a degree of Bachelor Engineering major in manufacturing and production from Technological University of the Philippines Taguig in 2021. She is currently an assistant PCB design engineer in ROHM LSI Design Philippines Incorporation. Her research focuses on PCB design engineering and component library systems development. She contacted at email: machallina.cuntapay@tup.edu.ph.

Mark Joseph B. Enojas received a degree of Bachelor of Science in Electronics and Communications Engineering from Technological University of the Philippines Taguig in 2009, Master of Information Technology in University of the Philippines Los Baños, Laguna, Philippines and is currently pursuing Ph.D. in Electrical and Electronics Engineering in University of the Philippines Diliman, Quezon City Philippines. His research interests are on mixed signals systems, mechatronics, industrial automation, and soft robotics for biomedical applications. He contacted at email: markjoseph_enojas@tup.edu.ph.

Hohn Lois C. Bongao received a degree of Bachelor of Science in Electronics and Communications Engineering from Technological University of the Philippines Taguig in 2017. He is currently pursuing his Master of Science in Electronics Engineering in Polytechnic University of the Philippine. His research interests are on instrumentation, mechatronics, and industrial automation. He contacted at email: hohnlois_bongao@tup.edu.ph.