

Towards Unsupervised SEM Image Segmentation for IC Layout Extraction

Nils Rothaug Max Planck Institute for Security and Privacy Bochum, Germany

> Sinan Böcker Bundeskriminalamt Wiesbaden, Germany

Simon Klix Max Planck Institute for Security and Privacy Bochum, Germany

Endres Puschner Max Planck Institute for Security and Privacy Bochum, Germany Nicole Auth Bundeskriminalamt Wiesbaden, Germany

Steffen Becker Ruhr University Bochum Bochum, Germany Max Planck Institute for Security and Privacy Bochum, Germany

Christof Paar Max Planck Institute for Security and Privacy Bochum, Germany

ABSTRACT

This paper presents a novel approach towards unsupervised SEM image segmentation for IC layout extraction. Existing methods typically rely on supervised machine learning with manually labeled training data, requiring re-training and partial annotation when applying them to new datasets. To address this issue, we propose a SEM image segmentation algorithm based on unsupervised deep learning, eliminating the need for manual labeling. We train and evaluate our approach on a real-world dataset comprising 648 SEM images of metal-1 and metal-2 layers from a commercial IC, achieving competitive segmentation error rates well below 1%. Releasing our dataset and algorithm implementations, we allow researchers to apply our approach to their own datasets and evaluate their methods against our dataset, facilitating reproducibility in the field.

CCS CONCEPTS

• Security and privacy \rightarrow Hardware reverse engineering.

KEYWORDS

IC layout extraction, SEM image segmentation, unsupervised deep learning, open-source dataset

ACM Reference Format:

Nils Rothaug, Simon Klix, Nicole Auth, Sinan Böcker, Endres Puschner, Steffen Becker, and Christof Paar. 2023. Towards Unsupervised SEM Image Segmentation for IC Layout Extraction. In *Proceedings of the 2023 Workshop on Attacks and Solutions in Hardware Security (ASHES '23), November 30, 2023, Copenhagen, Denmark.* ACM, New York, NY, USA, 6 pages. https: //doi.org/10.1145/3605769.3624000



This work is licensed under a Creative Commons Attribution International 4.0 License.

ASHES '23, November 30, 2023, Copenhagen, Denmark © 2023 Copyright held by the owner/author(s). ACM ISBN 979-8-4007-0262-4/23/11. https://doi.org/10.1145/3605769.3624000

1 INTRODUCTION

Integrated Circuit (IC) reverse engineering has a broad range of applications, including competetive analysis [18] and detecting counterfeits [14] or malicious circuit modifications [13]. Accurately extracting (parts of) the layout from an IC is an important task in this process [8]. For dies manufactured on modern technology nodes, it involves chemical and mechanical preparation and delayering of the chip prior to imaging each layer of the die with a Scanning Electron Microscope (SEM) [18]. Due to the usually imperfect SEM image quality, a major challenge of layout reconstruction is to segment all metal layers as precisely as possible into background, tracks, and vias [2]. State-of-the-art IC SEM image segmentation approaches often employ supervised machine learning [6, 17, 21], relying on manually annotated images to serve as labels during the training process. However, models trained on one dataset are often not directly applicable to others due to differing preparation, manufacturing, and imaging parameters [2]. Instead, models must be re-trained for each new dataset, which must either be partially annotated or otherwise preprocessed to fit the model's training data, causing a performance degradation [16]. The dataset differences make a fair comparison between segmentation methods almost impossible. Furthermore, some literature only reports pixel-wise evaluation metrics, which contain very little information about the actual segmentation quality in terms of electrically relevant errors. And while meaningful metrics, such as the Electrically Significant Difference (ESD)¹ have been proposed [9], the lack of open datasets and algorithm implementations obstructs a thorough comparison between segmentation algorithms for IC layout extraction. In this work, we strive to address these problems as follows:

• First, we devise an automated approach for track segmentation on metal-layer IC SEM images that eliminates the need for costly and time-consuming manual labeling. To this end,

¹While the acronym ESD usually refers to the term *electrostatic discharge*, we have chosen to remain consistent with the original work.

we present a novel algorithm based on unsupervised deep learning that – not relying on labeled training data – allows for adoption to new datasets with little human interaction.

- Second, we perform a thorough evaluation of our approach on a real-world dataset consisting of 648 SEM images from the metal-1 and metal-2 layers of a commercial 180 nm IC. Our results indicate low ESD error rates of down to .26%, which is competitive to state-of-the-art approaches for SEM image segmentation that use supervised learning.
- Third, we enable other researchers to compare their segmentation methods to our results and apply our method to their own datasets by making the data we used in our work available under an open-source license. This data consists of our algorithm implementations, our evaluation code, as well as the real-world SEM image dataset described above, including a corresponding ground truth for the metal-2 layer.

2 ARCHITECTURE FOR UNSUPERVISED SEM IMAGE SEGMENTATION

In this section, we briefly introduce the advantages of unsupervised learning and propose an unsupervised Machine Learning (ML) architecture for SEM image segmentation that allows adaptation to new datasets without prior manual labeling.

In recent years, deep learning techniques have achieved impressive results for image analysis tasks, such as segmentation [10]. A common issue for supervised ML approaches is the availability of labeled training data, which is why unsupervised and semisupervised approaches have gained traction. A supervised ML model learns a function that maps input data to provided labels. In contrast, an unsupervised model trains without labeled data and extracts information directly from the input distribution [1].

We base our architecture on the autoencoder design, which consists of two ML models, encoder and decoder [5]. The encoder compresses an input SEM image into a self-learned so-called *hidden representation*, from which the decoder reconstructs the original data. Our goal is to shape this hidden representation into a segmentation mask that classifies each pixel as either silicon background, metal track, or metal via. When training encoder and decoder together, the autoencoder would optimize its hidden representation for optimal input reconstruction, without forming a concept of background, track, or via segments. Instead, we train the decoder with segmentation masks obtained from conventional image segmentation algorithms, forcing the encoder to output a representation close to the decoder's learned input format.

To generate training data for the autoencoder, we denoise the SEM images in a preprocessing phase using median filtering and split them into 128×128 pixel SEM patches to improve scalability.

2.1 Conventional Segmentation Algorithms

For decoder training, we apply one of three conventional SEM image segmentation algorithms to the SEM image patches. Although their performance is inadequate for direct automatic IC layout reconstruction because they produce a large number of segmentation errors, they aptly shape the decoder's expected input and thereby constrain the autoencoder's hidden representation. **Fixed Threshold.** Arguably the simplest image segmentation algorithm, fixed thresholding classifies image pixels as either background, track, or via depending on their brightness. We pick minimum track and via brightnesses – the thresholds – based on visual inspection of the dataset and its image histograms.

Random Threshold. We also tested randomizing track and via thresholds on a per-patch basis by drawing them uniformly random distributed from an interval around the fixed thresholds.

Morphological Active Contours Without Edges. A more advanced segmentation algorithm, Morphological Active Contours Without Edges (MorphACWE) evolves a level-set curve along edges in the image while being resistant to noise [11]. We use fixed thresholds to obtain initial level-set curves and run the algorithm separately for track and via labeling.

2.2 Decoder

Figure 1a visualizes the decoder training. First, we apply the conventional segmentation algorithms from the previous section to our 128×128 pixel SEM image patches and thereby generate training data for the decoder. Instead of reconstructing the input SEM patch directly, we let the decoder predict an approximation of its gradient magnitude. The gradient magnitude of an image is the norm of the brightness differences between neighboring pixels in X and Y directions. Training the decoder to predict the image gradient, we prioritize the accurate placement of track and via boundaries over precise coloring of uniform areas, such as the background.

Reconstruction Loss. As reconstruction loss for decoder training, we use Mean Squared Error Loss (MSELoss). The loss compares the decoder output to ∇_{rec} *SEM* from Equation 1. In the equation, $\nabla_{\text{morph}} =$ dilation – erosion denotes the morphological gradient, an approximation of the gradient magnitude that, with a sufficiently large structuring element, produces less noise than the exact version. The hyperparameter λ_1 allows tuning the loss function in conjunction with clamping the gradient to the decoder output range. We choose $\lambda_1 = 5$, which saturates the gradients around vias and thus reduces the difference to the smaller gradients along track borders, balancing correct track border with via reconstruction.

$$\nabla_{\text{rec}} SEM = \text{clamp}_{[0,1]} (\lambda_1 \cdot \nabla_{\text{morph}} SEM)$$
(1)

As decoder architecture, we use U-Net [15] with batch normalization and sigmoid activation in the last layer for output normalization. Instead of deconvolution operations, we employ resizeconvolution to suppress high frequency artifacts in the output image [12]. We also use padded convolutions to retain the input size of 128×128 pixels for the output.

2.3 Encoder

The encoder receives SEM image patches and predicts their segmentation masks with separate channels for background, track, and via class probabilities. These masks are the primary output of our segmentation approach and serve as the basis for layout extraction.

In an unsupervised setting, we cannot assess the quality of the encoder output directly. We can, however, apply the decoder to the encoder output and compute the reconstruction loss of the resulting gradient prediction, as depicted in Figure 1b. Assuming that an accurate gradient reconstruction from the decoder requires a Towards Unsupervised SEM Image Segmentation for IC Layout Extraction



(a) A conventional segmentation algorithm generates masks from SEM image patches as training data. We train the decoder to predict a SEM patch gradient approximation from these masks using MSELoss.



(b) We train the encoder using the decoder and its reconstruction loss as encoder loss function, adding the channel exclusivity loss term L_{excl} to improve the generated segmentation.

Figure 1: Unsupervised training process of our encoderdecoder architecture.

high quality segmentation mask from the encoder, we gain an error metric for the encoder output. Propagating the reconstruction loss back through the decoder, we receive a differentiable loss function that allows us to train the encoder. During encoder training, we only update the encoder weights and do not train the decoder.

Class Exclusivity Loss. We add the term from Equation 2 to the encoder loss function to incentivize the encoder to predict that a pixel belongs exclusively to either background (*B*), track (*T*), or via (*V*) and scale this term by hyperparameter λ_2 , choosing $\lambda_2 = 0.1$. In conjunction with softmax, L_{excl} trains the encoder to predict mostly binary segmentation masks that allow for straightforward track and via extraction.

$$L_{excl} = P(B) P(T) + P(B) P(V) + P(T) P(V)$$
(2)

The encoder receives patches on the same 128×128 pixel grid as the decoder but with an added overlap to increase segmentation accuracy around patch edges, resulting in a 174×174 pixel input size. The architecture is similar to the decoder and also a U-Net, with the exception of partially using unpadded convolutions to shrink the larger input to the decoder patch size.

We train decoder and encoder alternately, which improves training stability compared to using a pre-trained decoder. When additionally training the decoder on encoder output, we find that the autoencoder optimizes its hidden representation for reconstruction, while losing its interpretability as segmentation masks. During deployment, we only require the encoder segmentation results and not the decoder, reducing complexity and size of the model.

3 A REAL-WORLD IC SEM IMAGE DATASET

To train and evaluate our IC image segmentation approach, we employ a real-world SEM image dataset, whose creation and characteristics we detail in this section. The data consists of 648 SEM images showing the logic area of the metal-1 (M1) and metal-2 (M2) layers of a commercial IC produced on a 180 nm technology node. The target IC has a total of six metal and a polysilicon layer and was developed for (medium) security applications. Figure 2 contains a sample image with ground truth from the M2 layer, which we labeled and used for our evaluation. We publish our dataset, including the M2 labels, under a permissive open-source license (CC-BY $4.0)^2$.

3.1 Chip Preparation and Imaging

Prior to imaging, we prepared the chip as follows: First, we chemically removed the device packaging, leaving the bare silicon die. The packaged chip dimensions are 3×3 mm and the actual die measures 2.16 mm×1.68 mm. After controlled removal of a thick aluminium (Al) top metal structure and a silicon nitride (SiN) / silicon dioxide (SiO₂) layer, we obtained an almost perfectly flat surface, which is the prerequisite for the following processing steps. Subsequently, we performed delayering of each metal interconnect layer using a broad ion milling system. Using argon (Ar) ions at a pressure of 4×10^{-4} mbar, 400 V beam voltage, and a current density of 0.18 mA/cm², we achieved an etch rate of 2 to 80 nm/min, depending on beam incident angle and material sputter yield.

Next, we acquired images of each layer using a Scanning Electron Microscope (SEM) with backscattered electron detector. As the backscatter electron yield is material dependent, both the metal layer and the underlying tungsten vias are visible in the images. To obtain satisfactory image contrast and signal-to-noise ratio, allowing us to discern tracks and vias from background, we used an electron energy of 15 keV and a dwell time of 3 µs.

3.2 Dataset Characteristics and Labeling

The M1 and M2 layers were captured with a resolution of 14.65 nm per pixel and 10% overlap between SEM images. Each image is 4096×3536 pixels in size. After discarding images of the area's boundaries, our dataset contains 327 images from the M1 layer and 321 from M2, yielding a total of 6 GiB grayscale image data.

Typically, the preparation and imaging processes create artifacts in SEM images that may cause challenges for segmentation. Uneven delayering, for example, yields inhomogeneous track and background coloring, which reduces contrast and, in extreme cases, can uncover tracks from adjacent metal layers. Additionally, bright spots that occur during imaging may interfere with via detection. Finally, vias in the SEM image are regularly surrounded by halos, which can make boundaries between neighboring tracks hard to determine, an artifact we call *bleeding*. Our dataset also contains such artifacts (see Figure 2a for an example).

We automatically labeled track polygons and via positions on the M2 layer, using techniques such as thresholding, edge detection, and size, position, and complexity filtering. In a final step, an experienced analyst manually validated and corrected these labels, which took an average of 6 min per image and a total of 6 person days for the entire layer. As the labels were originally intended for manual IC layout extraction, we decided against removing duplicates of detected vias. Furthermore, tracks spanning multiple images are not labeled on each image and instead appear as missing tracks when images are processed in isolation.

²https://doi.org/10.17617/3.HY5SYN



(a) Delayering artifacts manifest (b) Imperfect segmentation of (a) as uneven background and track containing a short in red and an coloring. Imaging artifacts are open track in blue. Track labels visible as bright spots on tracks. are outlined in gray.

Figure 2: Cutout of a SEM image (a) from our M2-layer dataset with polygon track labels and ESD error visualization (b).

3.3 Existing Datasets

To our knowledge, we are the first to publish a real metal-layer IC SEM image dataset and only two other open datasets exist. Cheng et al. [3] annotated gates on 640 polysilicon-layer IC SEM images with 384×512 pixels each, which are available on request. Wilson et al. [20] generated and published 800,000 synthetic polysilicon and M1-layer SEM images from 32 nm and 90 nm IC layout files. With 250×250 pixels each, the simulated images are much smaller than our real SEM images, only showing standard cell sections. Their simulation adds noise to emulate the imaging process, simple shape changes to imitate deformations from IC manufacturing, and shifted image regions to mimic stitching errors. In contrast to the artifacts we observed in our SEM images and described in the previous section, their background and tracks appear otherwise uniform.

4 EVALUATION

Here, we present the evaluation of our approach on our real-world dataset from the previous section. First, we advocate for meaningful evaluation metrics and propose using the Electrically Significant Difference (ESD) error metric introduced by Trindade et al. [9]. Second, we provide implementation details, report our results, and compare them to related approaches, where possible. Finally, we discuss limitations of our approach and future research directions.

4.1 Evaluation Metrics

IC layout extraction from SEM images is error prone, since even a few added or missing electrical connections between components can greatly affect the resulting layout and may require manual review and cleanup. Minimizing such faulty connections is thus paramount. However, commonly used evaluation metrics, namely mean Pixel Accuracy (mPA) and mean Intersection over Union (mIoU) [3, 4, 16, 19], assess the classification accuracy of individual pixels without considering connectivity. Inaccurately segmented track borders, while still allowing perfect layout recovery as long as connectivity remains unchanged, can therefore have a profound impact on mPA and mIoU. Conversely, bridging two tracks through a few incorrectly classified pixels induces a connectivity error, while having a negligible impact on the aforementioned per-pixel metrics. Therefore, these metrics do not provide a meaningful quality measure for SEM image segmentation with respect to layout extraction.

Electrically Significant Difference. For this reason, we evaluate our approach with the ESD metric proposed by [9], sometimes also referred to as Connected Component Analysis (CCA) [20]. This metric counts the number of electrical shorts and opens within the segmentation. Shorts are created when two distinct tracks are merged into one segment, and opens are single tracks that are split into multiple segments by the algorithm. We additionally report ESD false positives (FPs), i. e., segments without corresponding track in the ground truth, and false negatives (FNs), i. e., tracks that do not appear in the segmentation. Figure 2b illustrates short and open errors in a segmented sample image.

4.2 Implementation Details

While our approach does not require labeled training data, it relies on a few dataset-dependent parameters for the input segmentation algorithms, which we disclose below. We also detail the training and evaluation process and, along with our dataset (see Section 3), publish our implementation and trained models on GitHub³.

Conventional Segmentation Algorithms. As discussed in Section 2.1, all our conventional SEM image segmentation algorithms rely on some form of thresholding. We choose $\frac{67}{256}$ as track threshold and $\frac{157}{256}$ for vias. Deriving from a MorphACWE reference implementation⁴, we built a parallelized version of the algorithm using OpenCL. Based on qualitative visual inspection, we run MorphACWE for tracks and vias separately for 50 iterations each, with three rounds of smoothing per iteration and a foreground weight of 2 for tracks and $\frac{1}{2}$ for vias.

Training. We use PyTorch with Adam[7] optimizer and an initial learning rate of 2×10^{-4} , training our models for one epoch on a random subset of 200 SEM images from both the M1 and M2 layers of our dataset (see Section 3). Due to GPU memory constraints, we split each image into 8 batches, yielding a batch size of 108 128 × 128 pixel patches. Training takes approximately 25 min on a NVIDIA[®] Quadro RTX[™] 6000 GPU with 24 GB VRAM on a dual-socket server with Intel[®] Xeon[®] Gold 6154 CPUs and 376 GB RAM.

Evaluation. For performance reasons, and because our ground truth already consists of track polygons, we do not compute ESD errors on the segmentation mask directly. Instead, we binarize the mask before extracting track polygons using OpenCV and perform the ESD evaluation with polygon intersection. To reduce the number of false positives, we filter polygons with a total area smaller than 35 pixels and polygons enclosed by a 35 pixel margin around the image borders, due to missing labels (see Section 3.2).

4.3 Results

We evaluate the performance of our approach separately for all three segmentation algorithms, each of which providing training

³https://github.com/emsec/unsupervised-ic-sem-segmentation

⁴https://github.com/pmneila/morphsnakes



Figure 3: The graph shows the mean reconstruction loss and ESD error rate of all fifteen trained model instances. The error rate generally aligns with the reconstruction loss observed during training, allowing us to select well-performing models in an unsupervised setting. While some instances are outliers with a high ESD error rate, we can easily detect them based on their reconstruction loss. For comparison, we include the error rate of U-Net using supervised learning.

Table 1: Comparison of our approach with the input segmentation algorithms and the supervised U-Net trained directly on our evaluation data. We report the ESD errors from a total of 115,861 tracks on 321 M2-layer SEM images, as well as the mean Intersection over Union (mIoU) and mean Pixel Accuracy (mPA) as per-pixel metrics.

Alg.	ESD Errors					Pixel Metrics	
	Shorts	Opens	FPs	FNs	Total [%]	mIoU	mPA
Input*	11,063	41	463	0	9.984	0.805	0.948
Input†	2,005	26,480	247	39	24.832	0.747	0.940
Input‡	8,434	49	235	1	7.526	0.842	0.959
Ours*	282	154	296	1	0.633	0.838	0.958
Ours†	414	115	78	27	0.547	0.887	0.973
Ours‡	57	68	218	11	0.256	0.836	0.957
U-Net	189	96	33	98	0 359	0.900	0 977

*Fixed threshold. †Random threshold. ‡ MorphACWE.

data for the decoder (see Section 2.1). As we observe varying ESD error rates between runs, we train five instances of our model for every input segmentation algorithm. We select the instance with the lowest reconstruction loss, which tends to align with a high ESD performance (see Figure 3).

As ground truth for our evaluation, we use the manually labeled M2-layer data described in Section 3. Table 1 presents the ESD results of our models. As a lower benchmark, we report the performance of the three input segmentation algorithms when applied directly to our dataset. As an upper benchmark, we train our U-Net encoder on the labeled M2 layer data using supervised learning. The results show that our approach is able to generate high-quality segmentation masks in an unsupervised setting.

ESD Errors. Our approach achieves a total ESD error rate well below 1%, comparable to the U-Net we use as upper benchmark. The MorphACWE instance even outperforms the supervised model

on total ESD errors. All ESD metrics but the number of shorts, i. e., merged tracks, are consistent between most instances at a low rate. The random thresholding input algorithm is an expected exception and often splits tracks, producing a relatively large number of opens. The supervised U-Net yields the largest number of false negatives (FNs), i. e., undetected tracks, while generating less false positives (FPs) than the unsupervised algorithms. As a segmentation quality differentiator, our instances produce more than an order of magnitude fewer shorts than the conventional segmentation algorithms. From the results reported in Table 1 and from Figure 3, we infer that using MorphACWE as input algorithm improves training stability and, on average, performance compared to fixed or random thresholding.

Per-Pixel Metrics. As discussed in Section 4.1, we consider mIoU and mPA insufficient performance measures for segmentation quality in the context of layout extraction. We support our claim with our evaluation results and observe that all algorithms achieve similar per-pixel performances, while producing vastly different amounts of ESD errors. Particularly, the MorphACWE input segmentation algorithm performs better on both per-pixel metrics than our unsupervised ML approach using the same algorithm, while its ESD error rate is an order of magnitude higher.

4.4 Related Work

Here we place our results in the context of four ML approaches to track segmentation that also report ESD errors.

Hong et al. [6] train a Convolutional Neural Network (CNN) on 200 labeled SEM images and achieve 0.83 shorts respectively 0.26 opens per 2048×1536 pixel image. Our approach, for comparison, achieves between 1.103 and 2.283 errors per image with four times the pixel count. Using a Generative Adversial Network (GAN) on small labeled patches, Tee et al. [17] achieve with a mPA of 0.9442 and a mIoU of 0.8563 a similar per-pixel performance to our approach and report an ESD error rate of 4.71%, albeit on SEM images with a higher track density, based on their figures. Yu et al. [21] train a CNN on 21 8192×8192 pixel SEM images for 100 epochs and use post processing to reduce the number of ESD errors, reporting 50.71 errors total (2.381 per image) with 95.75% mPA and a high 91.86% mIoU. The images again appear to have a higher track density than our dataset. Wilson et al. [20] tested multiple ML and non-ML algorithms on their synthetically generated REFICS dataset (see Section 3.3), achieving segmentation error rates as low as 1% using CycleGAN[22] trained on labeled data, with a 89% mIoU.

Although not directly comparable due to the different underlying datasets, the accuracy of our approach appears to be on par with the state of the art, which, however, requires labeled training data.

4.5 Limitations and Future Research

Although our approach yields promising initial results, it still has limitations that necessitate further research in this area.

First, unsupervised learning is limited by the complexity of patterns it can observe in the input data which for our dataset – due to random bright artifacts caused by the imaging process – interferes with reliable via detection. While a supervised approach with extensive labeling might learn to differentiate such artifacts, our approach has the advantage of automatically generating a low-error segmentation for tracks without any manual annotations. In future research, unsupervised track segmentation algorithms could be combined with specialized algorithms for via detection [13].

Second, we employed three relatively simple conventional input segmentation algorithms for decoder training. In the future, evaluating other input segmentation algorithms could further improve our approach with respect to performance, stability during training, or generalizability to different datasets. Cheng et al. [3], for example, use multi-level Otsu's thresholding as a baseline, which chooses optimal thresholds based on the image histogram and would eliminate parameters to the input algorithm.

Third, we evaluated our approach on a single metal layer of our real-world SEM image dataset, where it performed well. Manual inspection of selected results on our dataset's M1 layer, which features much thinner and more densely packed tracks, indicates a deterioration in the segmentation quality of our approach. To evaluate the adaptability of our approach across node sizes and technologies, more and diverse SEM image datasets – including their ground truth – from different chip layers and nodes are necessary.

5 CONCLUSION

In this work, we have introduced a novel unsupervised approach for SEM image segmentation in IC layout extraction. Our method eliminates the need for manual labeling by leveraging unsupervised deep learning, enabling easier adaptation to new datasets. Our evaluation on the M2 layer of a real-world dataset demonstrated low Electrically Significant Difference (ESD) error rates for track segmentation, comparable to state-of-the-art supervised approaches. However, challenges for successful layout extraction remain, such as imaging artifacts and the diversity of materials and technologies used in IC fabrication. To address these challenges, future research could explore alternative input segmentation algorithms to improve performance and generalizability, or otherwise combine our approach with supervised or conventional image processing.

To foster reproducibility and facilitate further research, we will release our dataset and algorithm implementation under an opensource license. This will enable other researchers to apply our approach to their own datasets and evaluate their methods using our dataset, fostering fair comparisons among different segmentation algorithms. By promoting collaboration and transparency, we hope to drive progress in the field of IC layout extraction.

ACKNOWLEDGMENT

This work was supported by Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy – EXC 2092 CASA – 390781972 and by the German Federal Ministry of Education and Research (BMBF) under the project FINANTIA – 13N15298.

REFERENCES

- Yoshua Bengio, Aaron C Courville, and Pascal Vincent. 2012. Unsupervised feature learning and deep learning: A review and new perspectives. CoRR, abs/1206.5538 1, 2665 (2012), 2012.
- [2] Ulbert J. Botero, Ronald Wilson, Hangwei Lu, Mir Tanjidur Rahman, Mukhil A. Mallaiyan, Fatemeh Ganji, Navid Asadizanjani, Mark M. Tehranipoor, Damon L. Woodard, and Domenic Forte. 2021. Hardware Trust and Assurance through Reverse Engineering: A Tutorial and Outlook from Image Analysis and Machine

Learning Perspectives. ACM J. Emerg. Technol. Comput. Syst. 17, 4 (2021), 62:1-62:53.

- [3] Deruo Cheng, Yiqiong Shi, Tong Lin, Bah-Hwee Gwee, and Kar-Ann Toh. 2021. Delayered IC image analysis with template-based Tanimoto Convolution and Morphological Decision. *IET Circuits, Devices & Systems* 16, 2 (Aug 2021), 169–177.
- [4] Deruo Cheng, Yiqiong Shi, Tong Lin, Bah-Hwee Gwee, and Kar-Ann Toh. 2018. Hybrid K -Means Clustering and Support Vector Machine Method for via and Metal Line Detections in Delayered IC Images. *IEEE Transactions on Circuits and* Systems II: Express Briefs 65, 12 (Dec 2018), 1849–1853.
- [5] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. 2016. Deep Learning. MIT Press, Cambridge, MA, USA, Chapter 14, 499–523. http://www.deeplearningbook. org.
- [6] Xuenong Hong, Deruo Cheng, Yiqiong Shi, Tong Lin, and Bah Hwee Gwee. 2018. Deep Learning for Automatic IC Image Analysis. In 2018 IEEE 23rd International Conference on Digital Signal Processing (DSP). IEEE, Shanghai, China, 1–5.
- [7] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings. San Diego, CA, USA.
- [8] Bernhard Lippmann, Ann-Christin Bette, Matthias Ludwig, Johannes Mutter, Johanna Baehr, Alexander Hepp, Horst Gieser, Nicola Kovač, Tobias Zweifel, Martin Rasche, et al. 2022. Physical and functional reverse engineering challenges for advanced semiconductor solutions. In *Design, Automation & Test in Europe Conference & Exhibition (DATE)*. IEEE, Antwerp, Belgium, 796–801.
- [9] Bruno Machado Trindade, Eranga Ukwatta, Mike Spence, and Chris Pawlowicz. 2018. Segmentation of Integrated Circuit Layouts from Scan Electron Microscopy Images. In 2018 IEEE Canadian Conference on Electrical & Computer Engineering (CCECE). IEEE, Quebec, QC, Canadas, 1–4.
- [10] Shervin Minaee, Yuri Boykov, Fatih Porikli, Antonio Plaza, Nasser Kehtarnavaz, and Demetri Terzopoulos. 2022. Image Segmentation Using Deep Learning: A Survey. IEEE Transactions on Pattern Analysis and Machine Intelligence 44, 7 (2022), 3523–3542.
- [11] Pablo Márquez-Neila, Luis Baumela, and Luis Alvarez. 2014. A Morphological Approach to Curvature-Based Evolution of Curves and Surfaces. IEEE Transactions on Pattern Analysis and Machine Intelligence 36, 1 (2014), 2–17.
- [12] Augustus Odena, Vincent Dumoulin, and Chris Olah. 2016. Deconvolution and Checkerboard Artifacts. http://distill.pub/2016/deconv-checkerboard
- [13] Endres Puschner, Thorben Moos, Steffen Becker, Christian Kison, Amir Moradi, and Christof Paar. 2023. Red Team vs. Blue Team: A Real-World Hardware Trojan Detection Case Study Across Four Modern CMOS Technology Generations. In *IEEE Symposium on Security and Privacy (SP)*. IEEE Computer Society, Los Alamitos, CA, USA, 56–74.
- [14] Shahed E Quadir, Junlin Chen, Domenic Forte, Navid Asadizanjani, Sina Shahbazmohamadi, Lei Wang, John Chandy, and Mark Tehranipoor. 2016. A Survey on Chip to System Reverse Engineering. ACM journal on emerging technologies in computing systems (JETC) 13, 1 (2016), 1–34.
- [15] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. 2015. U-Net: Convolutional Networks for Biomedical Image Segmentation. In Medical Image Computing and Computer-Assisted Intervention – MICCAI. Springer International Publishing, Munich, Germany, 234–241.
- [16] Yee-Yang Tee, Deruo Cheng, Chye-Soon Chee, Tong Lin, Yiqiong Shi, and Bah-Hwee Gwee. 2022. Unsupervised Domain Adaptation with Histogram-gated Image Translation for Delayered IC Image Analysis. In 2022 IEEE Physical Assurance and Inspection of Electronics (PAINE). IEEE, Huntsville, AL, USA, 1–7.
- [17] Yee-Yang Tee, Xuenong Hong, Deruo Cheng, Chye-Soon Chee, Yiqiong Shi, Tong Lin, and Bah-Hwee Gwee. 2023. Patch-Based Adversarial Training for Error-Aware Circuit Annotation of Delayered IC Images. *IEEE Transactions on Circuits* and Systems II: Express Briefs 70, 9 (2023), 3694–3698.
- [18] Randy Torrance and Dick James. 2009. The State-of-the-Art in IC Reverse Engineering. In Workshop on Cryptographic Hardware and Embedded Systems. Springer Berlin Heidelberg, Lausanne, Switzerland, 363–381.
- [19] Ronald Wilson, Domenic Forte, Navid Asadizanjani, and Damon L. Woodard. 2020. LASRE: A Novel Approach to Large area Accelerated Segmentation for Reverse Engineering on SEM images. In *International Symposium for Testing and Failure Analysis*. ASM International, Online, 180–187.
- [20] Ronald Wilson, Hangwei Lu, Mengdi Zhu, Domenic Forte, and Damon L. Woodard. 2022. REFICS: A Step Towards Linking Vision with Hardware Assurance. In 2022 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV). IEEE, Waikoloa, HI, USA, 3461–3470.
- [21] Zifan Yu, Bruno Machado Trindade, Michael Green, Zhikang Zhang, Pullela Sneha, Erfan Bank Tavakoli, Christopher Pawlowicz, and Fengbo Ren. 2022. A Data-Driven Approach for Automated Integrated Circuit Segmentation of Scan Electron Microscopy Images. In 2022 IEEE International Conference on Image Processing (ICIP). IEEE, Bordeaux, France, 2851–2855.
- [22] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. 2017. Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks. In 2017 IEEE International Conference on Computer Vision (ICCV). IEEE, Venice, Italy, 2242–2251.