INTEGRATING CONDITIONAL SHAPE EMBEDDING WITH GENERATIVE ADVERSARIAL NETWORK -TO ASSESS RASTER FORMAT ARCHITECTURAL SKETCH

Han Tu

Department of Architecture
Massachusetts Institute of Technology
77 Massachusetts Avenue,
Cambridge, MA, USA
hantu@mit.edu

Yichao Shi

School of Architecture
Georgia Institute of Technology
245 4th St NW,
Atlanta, GA, USA
yshi431@gatech.edu

Meng Xu

Department of Architecture
Aalto University
Otakaari 1B,
02150 Espoo, FINLAND
meng.xu@aalto.fi

ABSTRACT

Automatic assessments of building plans are uncommon in the early design stages, especially when schematic sketches are in raster format. Existing design evaluation tools, such as fire code reviewers, primarily evaluate vector format images that contain complete building information in the late design stage. These tools use conditional shape-embedding techniques to analyze the vector images. However, there are limitations to identifying and evaluating drawings through vector-shape relationships. Our research aimed to develop tools that can automatically assess schematic sketches in raster format to overcome the limitations of existing tools. We integrated a conditional shape-embedding tool, named Shape Machine, to assess vector images, with machine learning techniques to assess raster sketches. This integration enables the evaluation of fire evacuation sketches in the early stages of the design process, thereby improving design efficiency and reducing costs. Moreover, in the future, this integration could allow the evaluation of designs in multiple image formats.

Keywords: Machine Learning, Shape Grammar, Design Assessment, Fire Codes, Generative Adversarial Network.

1 INTRODUCTION

When conducting detailed design, architects use various drawing tools, including schematic sketches, which often require file format conversion. But this conversion of sketches from raster to vector format is a time-consuming process that can hinder the automatic review of fire codes in the early stages of design.

Despite the widespread use of vector image technology for detailed design and image manipulation, designers still prefer sketching in raster formats to express creativity during the early stages of visual invention (Fish and Scrivener 1990). However, existing design evaluation systems, such as VAIplus, SMARTreview APR, and CivitPERMIT, are limited to analyzing vector-format files, which are usually not available until later design phases, thus leading to a lack of early-stage review of sketches that can cause costly delays and expenses (Rounce 1998). In addition, late-stage building plan adjustments for fire codes typically require considerable collaborative revision time involving other related engineering inputs, such as structural and material drawings. As a result, there is a need for automatic tools that can review compliance with local codes during early design phases and adapt to universal formats such as JPEG, unlike existing fire evacuation review tools, which mainly focus on complex plans or models in later design stages.

Our research aims to address these efficiency and image format constraints associated with building a sketch plan assessment for fire code criteria in the early stages of design. To assess raster format images in early design stages, we integrated Shape Machine, a conditional shape-embedding tool (Economou et al. 2021), with a Generative Adversarial Network (GAN). We accomplished this integration by identifying vector lines using the Shape Machine as a collection of raster colors, rendering the raster images readable by the GAN. To evaluate this the proposed methodology as a general-purpose tool to assess building sketches for different applications, we utilized the fire code checks in an early stage as a hypothetical study case.

In our research, we review early-stage drawings in JPEG format for compliance with fire code criteria using image-processing technology. To accomplish this, we utilized the Code for Fire Protection Design of Buildings, GB 50016-2014 (2018 Edition), and floor plans of schools as examples. Our approach involved several steps. First, we decomposed the local fire code into six important types, which we marked with different colors in the early-stage sketches. Second, we translated the fire code criteria into code statements, particularly conditional shape-embedded rules, to enable the review of the key elements of the building sketch plans. Third, we used a newly developed shape detection tool based on shape grammar, called Shape Machine, to recognize the key elements and mark them as vectors in school plans. We marked the room vector lines recognized in the raster image sketches with different colors corresponding to various items in the fire code criteria. These raster images were then used for training the GAN. Specifically, we built a dataset of 300 paired images as input for training. Fourth, we trained the pix2pix GAN using the paired raster fire code assessment dataset color-marked by the Shape Machine. Thus, the trained model produces a tool for automatically reviewing architectural sketches.

Our research revealed that the resulting network accurately recognized certain sketch plan features, including the distances between doors and exits. In addition, we compared the efficiency of our trained results with those checked by humans and observed that our network was capable of enhancing the efficiency of the early-stage fire code review process for sketches in the raster format.

Furthermore, the study demonstrated that the Shape Machine can effectively extract the key image features from building sketch plans, whereas the GAN can assess the original sketches in raster format. The combination of both tools enables the recognition of incorrect survival distances and element sizes from images in raster format. Thus, the integration offers a more objective means of evaluating the compliance of early-stage design ideas with fire code standards. The use of Shape Machine and GAN machine-learning technologies in evaluating fire evacuation strategies during early sketch design phases can significantly increase design efficiency and reduce costs. In addition, this integration has the potential to enable future design assessments using various picture formats. Our code and dataset can be accessed through https://doi.org/10.2023/fire.code.

2 RELATED WORK

In the early stages of design, such as the schematic sketching stage, designers can benefit from using a wider range of tools to explore design diversity (Bueno and Turkienicz 2014; Pranovich 2004). However, this diversity can lead to various design file formats in those early stages. One possible solution to this issue is to review fire code criteria compliance using universal image formats that all software can export, such as JPEG. Using universal formats as input for reviewing plans can expand designers' capabilities in the early stages of sketch design.

Existing building code review tools are typically designed to review vector-format plans in the late stages of design. For example, VAIplus can only check plans for fire code requirements during the design development and construction document phases using platforms that contain detailed information, such as building information modeling (BIM) and AutoCAD (Ismail, Ali, and Iahad 2017). Therefore, an automated review tool can enhance design efficiency with flexible requirements for universal format input in the initial stages of sketch design.

Furthermore, existing fire code review tools, such as VAIplus, SMARTreview APR, and CivitPERMIT, are limited in terms of the review stages, review requirements, and file formats (as shown in Table 1). These tools typically require detailed layer-by-layer information, including plumbing and electrical information, which is often not available until late in the design process, resulting in costly modifications. In contrast, during the early stages of design, information only exists regarding the relationship between space division and room function, but not about material and structural layers. As a result, during the early stages of sketch design, when modifications can be made more cost-effectively, existing fire code tools are unable to review plans or provide suggestions. Thus, in the future, automatic review of early sketches could potentially reduce the cost of later modifications by engineers of different professions (Rounce 1998).

Table 1: Existing fire code review tools.

Checking Phase	Platforms	Fire Code Criteria	Format	Sources
Design development /Construction documents	VAIplus	Codes for architectural, structural, and electromechanical design (China)	CAD	vaiplus.com
	CivitPERMIT	Codes for architectural, structural, and electromechanical design (USA)	CAD, BIM	thecivit.com
	SMARTreviewAPR	International Building Code (IBC)	BIM	smartreview.biz
Construction documents	Glodon BIM Reviewer	Codes for architectural, structural, and electromechanical design (China)	BIM	glodon.com
	PKPM	Codes for structural design (China)	CAD, BIM	pkpm.cn
	xzst360	Codes for structural design (China)	CAD	xzst360.com
Schematic design /Design development/ Construction documents	TUZHI.AI	Fire codes for residential, public, and industrial buildings (China)	pdf drawings	tuzhi.ai

We have integrated Shape Machine (Economou et al. 2021) and a pix2pix GAN (Goodfellow et al. 2014) to overcome the limitations of existing fire code review tools in assessing floor plans according to local building requirements. Shape Machine is a general shape grammar interpreter that follows the fluent eye-hand, seeing-doing workflows emphasized by shape grammar formalism for vector images (Economou et al. 2021). It recognizes vectors, which can then be marked with raster colors for neural network recognition. The pix2pix GAN (Isola et al. 2018) is a machine learning tool that solves image-to-image translation problems in raster format with a minimal training set and supervised learning performance. GANs have become a well-established method in various types of design assessment, including architectural sketch generation (Qian, Xu, and Li 2022), architectural plan generation (Chaillou 2020), urban visual quality assessment (Guo et al. 2020), environment simulation (Mokhtar, Sojka, and Davila 2020), performative design (Duering, Chronis, and Koenig 2020; Lorenz 2019), and design generation and recognition (Huang 2018). Image-processing technology that utilizes machine-learning tools can assess plans and models in various architectural fields.

However, studies on the use of raster image-processing technology with vector recognition grammar for sketch assessment, particularly with respect to fire code criteria, are scarce in the literature. By combining a pix2pix GAN and Shape Machine, evacuation sketches in JPEG format could be evaluated disregarding the vector format prerequisites and complex requirements of later design stages. Therefore, in this study, we examined the efficacy of the Shape Machine and pix2pix GAN in evaluating early-stage building sketches for fire code criteria.

3 METHODOLOGY

To evaluate the effectiveness of the Shape Machine and GAN integration in assessing raster format sketches for fire code criteria compliance, we utilized school floor plan sketches as examples. As shown in Figure 1, our methodology consists of four steps: plan recognition, fire code translation, image format transition for dataset creation, and GAN training.

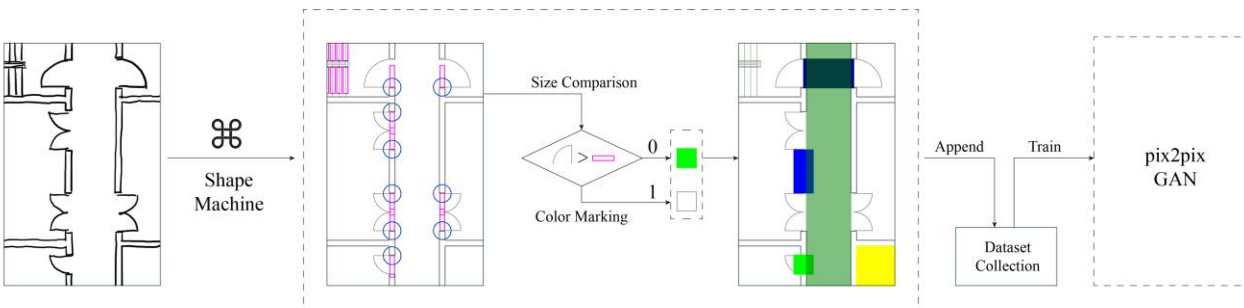

Figure 1: Workflow of the proposed methodology involving: plan recognition, fire code translation, image format transition for dataset creation, and GAN training.

3.1 Plan Recognition

Fire codes vary depending on the type of building, such as residential, medical, and industrial. As a case study, we selected five-story schools with Class 1 fire resistance, no automatic sprinkler system, closed stairways and indoor corridors, and 250 persons per floor. We identified five key elements of school plans in accordance with the criteria in the Code for Fire Protection Design of Buildings, GB 50016-2014 (2018 Edition). These five elements included the distances between doors or exits (Code 1), the number of doors (Code 2), the distance between room doors and exits (Code 3), the evacuation distance in each room (Code 4), and the width of doors, exits, and corridors (Code 5). These elements were recognized as vectors using AutoCAD Raster Design Toolset, and their relationships were identified using Shape Machine (Economou et al. 2021).

Table 2: Example fire codes for checking function translation, cited from Code for Fire Protection Design of Buildings, GB 50016-2014 (2018 Edition).

Code No.	**Description**	**Item**
Code 1	[1.1] The horizontal distance between the nearest edges of two adjacent evacuation doors in each room shall not be less than 5 m. [1.2] The horizontal distance between the nearest edges of two adjacent safety exits on each floor in each fire compartment in the building shall not be less than 5 m.	5.5.2

Code 2	The number of evacuation doors in a room should be determined by calculation and should not be less than 2. For a room located between two safety exits or at both sides of the terminal of a dead-end, if the floor area is not larger than 75 m^2, an evacuation door can be set.	5.5.15
Code 3	The linear distance from the evacuation door to the nearest safety exit of the room directly leading to the evacuation corridor shall not be greater than those specified in Table 5.5.17: [22 m].	5.5.17.1
Code 4	The maximum linear distance from any point in the room to the evacuation door directly leading to the evacuation corridor should not be greater than the linear distance from the evacuation doors at two sides of the terminal of the dead-end to the nearest safety exit, as specified in Table 5.5.17: [22 m]	5.5.17.2
Code 5	The clear width of the evacuation doors (5.1) and safety exits (5.2) should not be less than 0.9 m. The clear width of the evacuation corridors (5.3) and evacuation stairways (5.4) should not be less than 1.1 m. The aggregate clear width of evacuation doors (5.1), safety exits (5.2), evacuation corridors (5.3), and evacuation stairways (5.4) on each floor shall be determined according to the calculation of the minimum clear evacuation width for every 100 persons but not be less than those specified in Table 5.5.21-1. For a building with fire resistance Class I and II, if the aboveground story is not less than 4, the minimum clear evacuation width is 1.00 m/hundred persons. [In our case, the minimum clear width is 2.5 m for 250 persons per floor per fire compartment, 1 in total.]	5.5.18 5.5.21.1 5.5.21.2
Code 6	When the number of floors does not exceed four and the first floor does not adopt an enlarged enclosed stairwell or smoke-proof stairwell foreroom, the distance between the door leading to the outside and the stairwell should not be more than 15 m for the first floor.	5.5.17.2

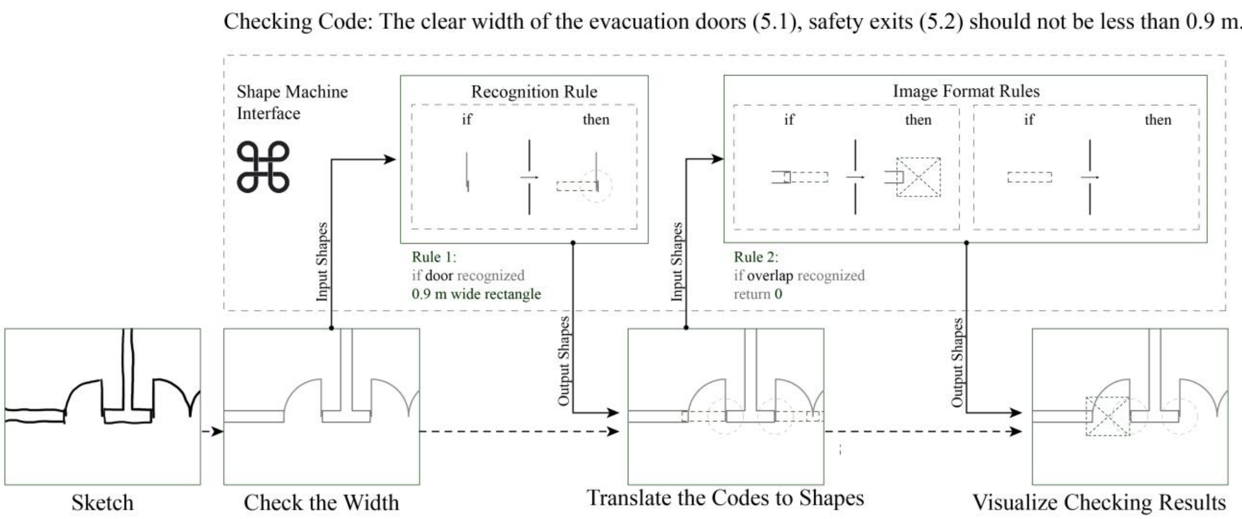

Figure 2: Sketch plan recognition, fire code translation, and image format transition using Shape Machine.

3.2 Fire Code Translation

We used Shape Machine (Economou et al. 2021), a technology based on shape grammar, to translate the fire codes into code statements and review the five key elements in building plans in vector format.

To build our dataset, the text in the fire code was translated into ten checking functions that help recognize vector image data in the generated plans and review their compliance with different fire code criteria (see Figure 2). We developed a labeling rule that used different colors to mark areas with different plan errors (see Figure 3). Colors with RGB values ranging from 0 to 255 were used to differentiate the labels as much as possible. Therefore, we used ten combinations of RGB values to label the ten types of errors corresponding to the different fire code criteria. If any element in the room did not meet the code requirements, the room was colored accordingly; for example, Room 1 in Figure 3 was labeled red (R:255 G:0 B:0) when breaking Code 4; similarly, four tones of green (R:0 G:63 B:0, R:0 G:63 B:0, R:0 G:63 B:0, and R:0 G:63 B:0) were used to label width errors when breaking Code 5, setting these drawing layers always on the top of the others.

We used Shape Machine (Economou et al. 2021) to translate each code in Table 2 into the checking functions listed in Table 3 for the recognition process. As depicted in Figure 4, Checking Functions 1.1 and 1.2 were used to examine whether the distance between two doors of a room or two exits was less than 5 m. Checking Function 2 was employed to verify if there were at least two doors when the room area was greater than 75 m^2. Checking Function 3 was utilized to check whether the shortest distance between doors and exits, as calculated through the Grasshopper Shortest Walk plug-in, was less than 22 m. Checking Function 5 evaluated whether the distance between points and doors in a room was less than 22 m. Specifically, Checking Functions 5.1, 5.2, 5.3, and 5.4 were utilized to review the minimum widths of doors, exits, corridors, and stairways, respectively. Finally, Checking Function 6 determined if the distance between the doors and exits of the first floor was less than 15 m. To complete all the checking functions, we utilized Shape Machine to recognize the vector-embedded shapes of the five elements and substitute them with raster colors. Subsequently, the reviewed elements were automatically color-coded in raster format to build a machine-learning image dataset.

Machine translates the raster to vector for vector recognition. However, Shape Grammar can be used to check the drawings of vector formats based on rule-by-rule checking. This process is not only time consuming, but also demands significant computing. For example, if a vector-format drawing includes many details, such as doors, stairs, or decorations that need to be checked, the Shape Machine—the interpreter of shape grammar—would have to go through all the lines. In this case, checking multiple rules of a floor plan would take significant time and computing, and reviewing the floor plan of a big building could take several hours. Therefore, to make the reviewing process more efficient we use GAN to check the raster drawings, which ignores the excessive vector information and enables checking multiple rules simultaneously. Moreover, to review raster sketches more efficiently, we integrate Shape Machine and GAN.

Figure 3: Input image and checked image with color marks for nine checking functions.

Table 3: Translation of selected codes from Table 2 into ten checking functions.

Code No.	**Checking Functions**	**Marked Colors**	**Checked Elements**
Code 1.1	`def function_1_1(x):` `return(1 if distance(x.door[0], x.door[1]) < 5 else 0)`	R:0 G:0 B:255	Distance between doors
Code 1.2	`def function_1_2(exit):` `return(1 if distance(exit[0], exit[1]) < 5 else 0)`	R:0 G:0 B:127	Distance between exits
Code 2	`def function_2(x):` `return(1 if num(x.door)<=1 and area(x) > 75 else 0)`	R:255 G:255 B:0	Number of doors
Code 3	`def function_3(x):` `return(1 if distance(x.door, exit) > 22 else 0)`	R:255 G:0 B:255	Distance between doors and exits
Code 4	`def function_4(x):` `return(1 if distance in point_to(x.door) > 22 else 0)`	R:255 G:0 B:0	Evacuation distance in a room
Code 5.1	`def function_5_1(x):` `return(1 if width(x.door) < 0.9 else 0)`	R:0 G:255 B:0	Door width
Code 5.2	`def function_5_2(exit):` `return(1 if clear_width(exit.door) < clear_width(exit.stairways) or clear_width(exit.door) < 1.1 or sum(clear_width(exit.door)) < 2.5 else 0)`	R:0 G:183 B:0	Exit width
Code 5.3	`def function_5_3(corridor):` `return(1 if clear_width(corridor) < 2.5 else 0)`	R:0 G:127 B:0	Corridor width
Code 5.4	`def function_5_4(stairways):` `return(1 if sum(clear_width(stairways)) < 2.5 or clear_width(stairways) < 1.1 else 0)`	R:0, G:63, B:0	Stairway width
Code 6	`def function_6(x):` `return(1 if distance(exit.door, 1F exit) > 15 else 0)`	R:0 G:255 B:255	Distance between doors of stairways and first floor exits

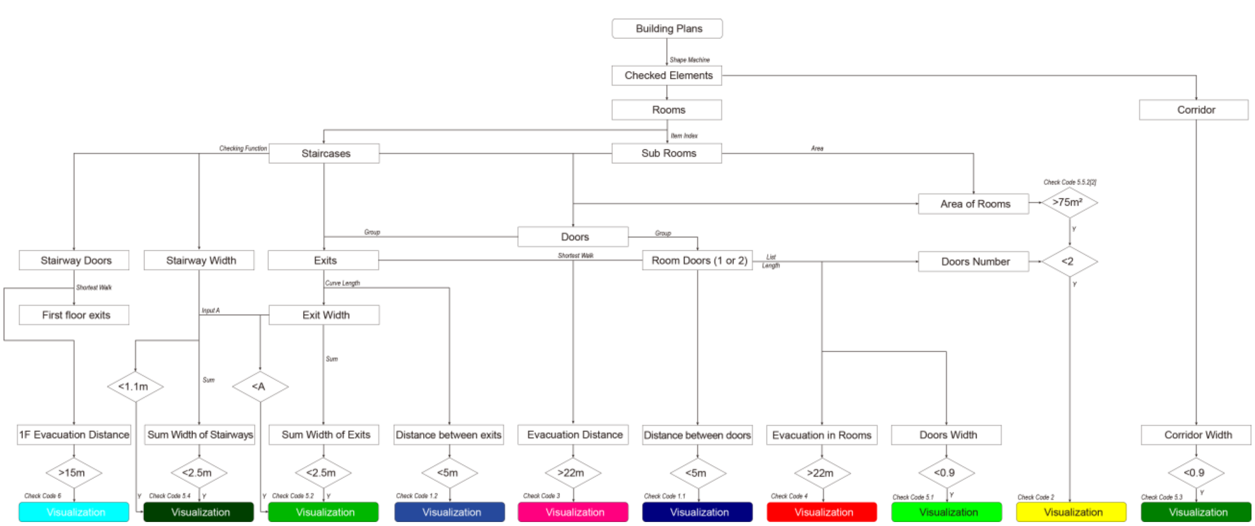

Figure 4: Ten checking functions and their corresponding colors.

3.3 Image Format Transition for Dataset Creation

The Shape Machine and the ten checking functions were utilized to review whether the five key elements in the vector format images conformed to the fire code criteria. Subsequently, the reviewed elements were converted into raster images to build an image dataset. The resulting dataset comprised 300 pairs of unreviewed and reviewed raster sketch images for machine learning purposes, as shown in Figure 6.

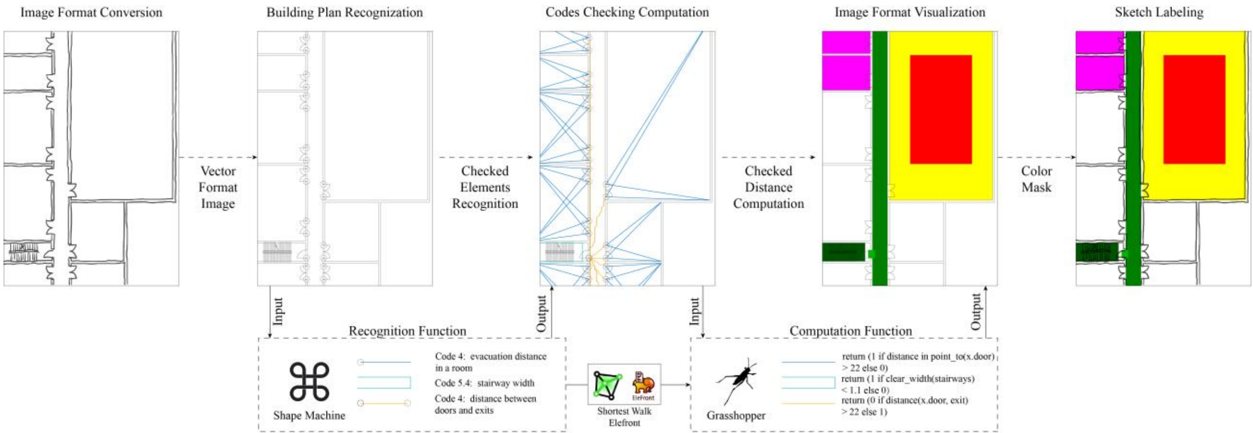

Figure 5: Procedure for recognizing vector information and translating it into raster formats using Shape Machine.

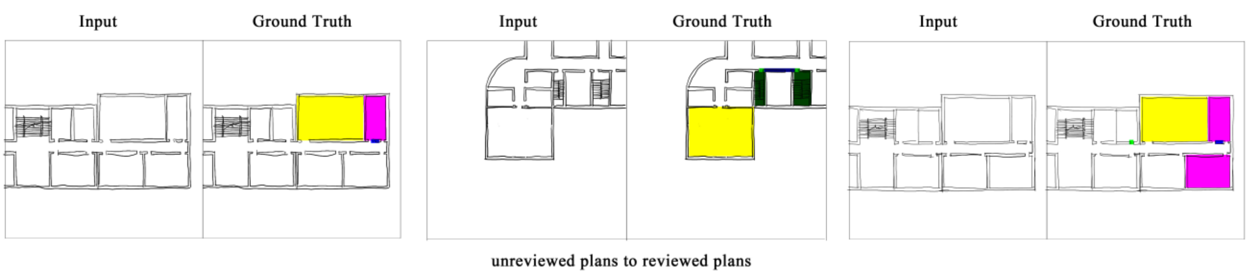

Figure 6: Dataset of unreviewed and reviewed building plan pairs for GAN training.

3.4 GAN Training

We used the fire code assessment dataset, color-marked by Shape Machine using all ten checking functions, to train the pix2pix GAN for converting unchecked sketches (virgin raster plans) to checked sketches (color-marked raster plans).

The dataset was divided into a training set of 250 images and a testing set of 50 images. The network was trained using plan sketches as inputs and color-marked sketch plans as outputs, i.e., the program generated a color-marked sketch plan indicating different errors based on a given sketch plan drawing. The entire training process was performed on Google Colab using an NVIDIA-SMI 460.32.03 GPU. One epoch with 250 images took 240 seconds and the full network training lasted 16.2 hours.

To validate the effectiveness of our approach, we compared the output produced by the GAN based on real-world school drawing inputs with the results obtained by manually applying each checking function Moreover, to analyze in detail the predictive efficiency and accuracy of the model, we trained the GAN using nine checking functions separately and asked three experienced designers to manually review each school plan within one minute to identify errors according to each fire-code criterion.. Subsequently, we averaged the efficiencies and accuracies of the experts and compared them to those of the GAN. The results are summarized in Table 4. The manual accuracy reported in the table indicates the average percentage of all non-compliant elements effectively identified by the experts in one minute.

4 EXPERIMENTAL RESULTS

By generating plans using all nine checking functions, the pix2pix GAN can create an image-to-image translation from an unreviewed raster sketch plan to a checked one with color-coded marked fire evacuation risks (Figure 7), resulting in a discriminator loss of 0.71 and generator loss of 1.82. We compared the efficiency of our trained model with that of manual checking and found that our network improved the efficiency of the fire code assessment process per plan by 30% within one minute.

To evaluate the recognition effect of different fire codes, we separately trained the pix2pix GAN using each of the nine checking functions. Some color markings from unreviewed to reviewed plans performed well in separate training, while others did not due to the varying geometric transform complexities of the plans. For codes 2, 3, and 4, both adversarial networks produced consistent and stabilized results (Figure 8). In contrast, for codes 5.1 and 5.2, the discriminator performed worse because it cannot easily distinguish between real (reviewed) and fake (unreviewed) plans (Figure 8). This can be attributed to the generator always trying to find the one output that seems most plausible to the discriminator. Therefore, while the GAN can correctly fill the room's original shape bits and sums and generate an appropriate color mask when evaluating codes like 2, 3, and 4, for codes 5.1 and 5.2, it needs to judge the evaluation method of color filling boundary by itself, which causes challenges in generating the corresponding masks.

The challenges encountered when marking colors separately help determine the fire code assessment checking functions in which either humans or machines perform better. Specifically, when we compare the machine learning results with manual ones, we find that machines excel at recognizing long distances and complex geometries, while experienced designers are more sensitive to small- and human-scale assessments (Table 4). For example, by breaking down the color-marked layers, we found that machine detection of codes 2, 3, and 4, which evaluate the number of doors, distances between exits and doors, and room sizes, was over 30% more accurate than manual evaluations. However, machine detection of codes 5.1 and 5.2, which evaluate the widths of doors and exits, was less sensitive than manual evaluations (Figure 8).

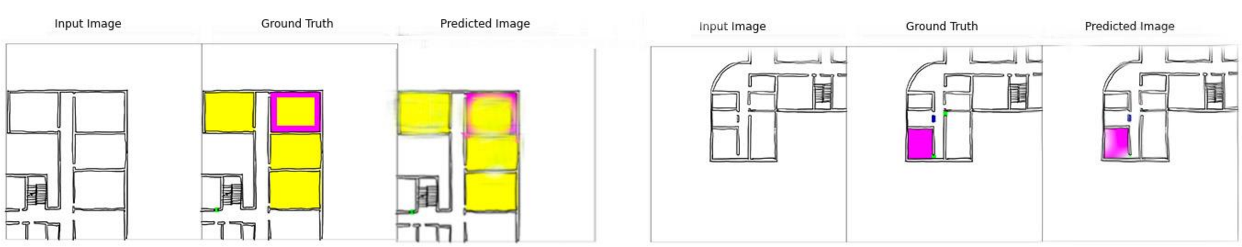

Figure 7: Results of pix2pix GAN transforming unreviewed to reviewed sketch plans with all layers.

Table 4: Training results and efficiency evaluation.

Code No.	Marked Colors	Checked Elements	GAN Accuracy	Manual Detected in 1 min
Total	Nine Colors	All elements	0.73	0.43
Code 1.1	R:0 G:0 B:255	Distance between doors	0.62	0.44
Code 1.2	R:0 G:0 B:127	Distance between exits	0.88	0.25
Code 2	R:255 G:255 B:0	Number of doors	0.46	0.61
Code 3	R:255 G:0 B:255	Distance between doors and exits	0.75	0.16
Code 4	R:255 G:0 B:0	Evacuation distance in a room	0.79	0.22

Code 5.1	R:0 G:255 B:0	Door width	0.85	0.61
Code 5.2	R:0 G:183 B:0	Exit width	0.91	0.21
Code 5.3	R:0 G:127 B:0	Corridor width	0.70	0.62
Code 5.4	R:0, G:63, B:0	Stairway width	0.64	0.75
Code 6	R:0, G:255, B:255	Distance between doors and exits	0.71	0.85

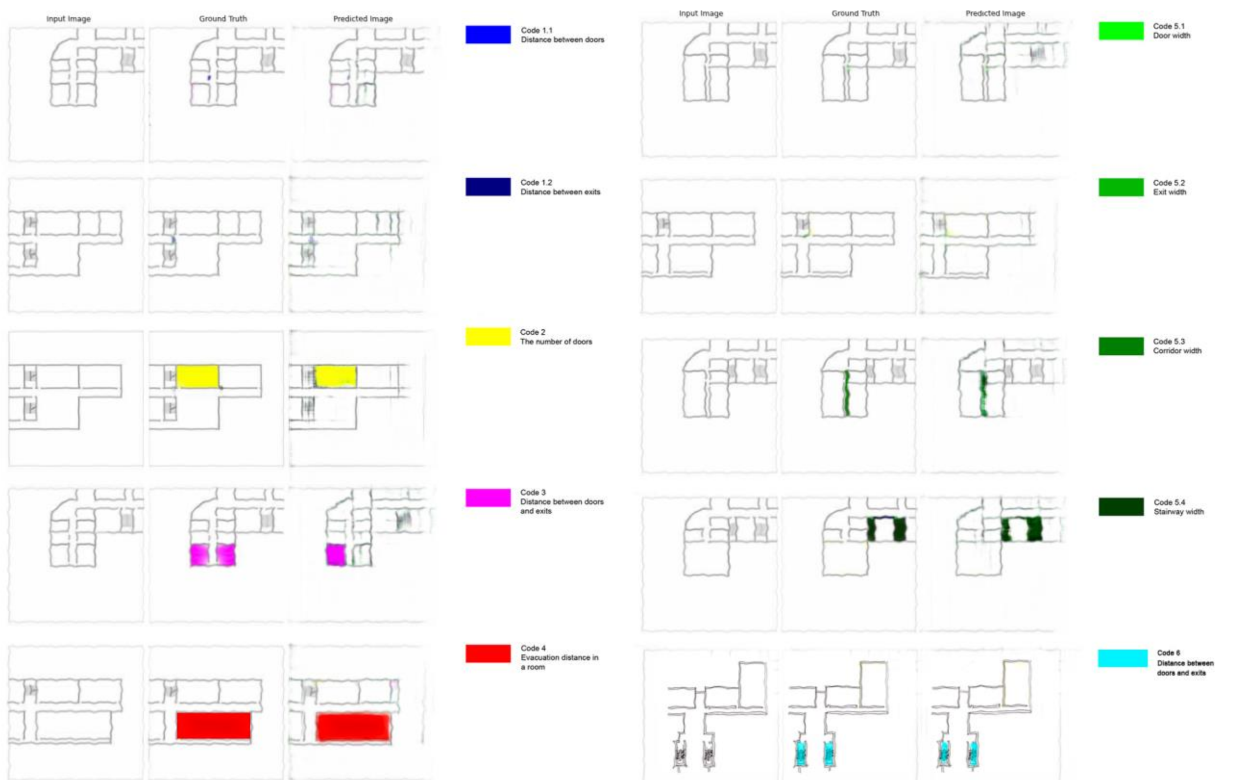

Figure 8: Results of pix2pix GAN transforming unreviewed to reviewed plans for different layers.

Finally, the trained GAN model was used to predict unseen floor plans to determine whether it could predict other sketching styles (Figure 9). The results showed that the model could provide a more efficient examination of different styles of plane sketches, indicating that the GAN integrated with the Shape Machine could more efficiently review geometrically complex school-plan features in the early sketch stages.

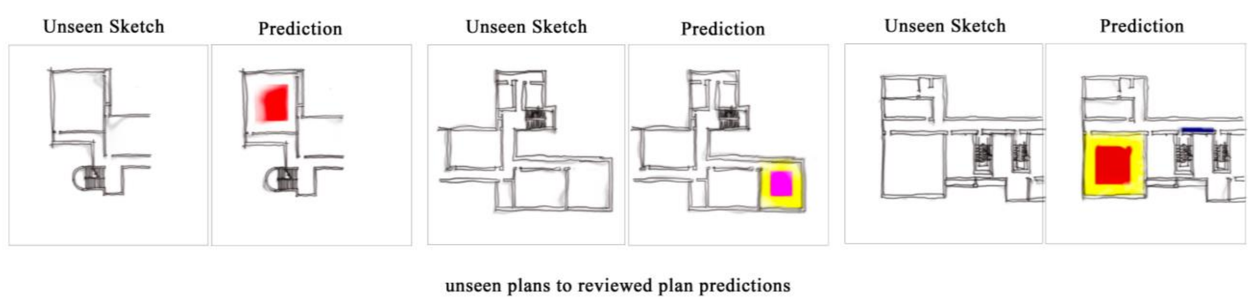

Figure 9: Results of the prediction of unseen sketches using our pix2pix GAN model.

5 CONCLUSIONS

This research proposes a new method that integrates Shape Machine and a GAN to assess raster building plans during the sketch design stage, specifically for reviewing plans according to local fire codes. As structured vector plans are not available in the early stages of design, we expanded our evaluation tools by using Shape Machine to recognize and mark key plan elements (Economou et al. 2021). The marked sketch plans can then be processed by the GAN to train a machine-learning model that automatically reviews the raster plans in the early stages of design. Our results indicate that this approach is effective for assessing raster images.

Moreover, this integration serves as a bridge between vector and raster images and between fire code and raster images for fire code reviews. We translated the sketch information into color-marked plans using fire-code checking functions and a Shape Machine. This translation bridges the gap between vector and raster information for a more effective review of universal-format sketches. This approach assists architects in reviewing plans with respect to local fire codes in the early stages of design, thereby improving design efficiency and reducing the cost of adjusting plans in later design stages.

Our research has limitations in terms of the size and domain biases of our dataset, and the relatively fixed assessment stage of the early sketch design stages. We tested only the floor plans of school buildings as an example, with a limited number of checking elements and sketch styles in the early stages of design, and selected translations of local fire codes. Our dataset contained 300 pairs of images, which limited the training and validation accuracy to 0.71. Thus, increasing the dataset size may improve the training and validation accuracy of the machine-learning model. Lastly, our experiment shows that a GAN integrated with a Shape Machine can be used as a rough method for evaluating sketch plans in the early schematic sketch design stage. However, the evaluation of complex sketch information has yet to be tested. Nonetheless, the proposed integration of Shape Machine and machine learning techniques provides novel ideas and approaches for image assessment in universal formats, particularly for the automatic evaluation of two-dimensional design sketches.

ACKNOWLEDGMENTS

This research was supported by the XKool Hackathon Marathon, 2022. We express our appreciation for their generous support. We would like to extend our sincere gratitude to the members of the Shape Computation Lab for their contributions to developing the Shape Machine. We would also like to thank Professor Athanassios Economou for his valuable insights and expertise in shape grammar. In addition, we are grateful to the Massachusetts Institute of Technology, Georgia Institute of Technology, and Aalto University for providing us with the resources and facilities necessary for this study.

REFERENCES

Bueno, E. , and Benamy T. 2014. "Supporting Tools for Early Stages of Architectural Design." International Journal of Architectural Computing 12 (4): 495–512.

Chaillou, S. 2020. "ArchiGAN: Artificial Intelligence x Architecture." In Architectural Intelligence: Selected Papers from the 1st International Conference on Computational Design and Robotic Fabrication (CDRF 2019), edited by Yuan, P.F., Xie, M., Leach, N., Yao, J., and Wang, X., 117–27. Singapore: Springer Nature.

Duering, S., Angelos, C., and Reinhard, K. 2020. "Optimizing Urban Systems: Integrated Optimization of Spatial Configurations." SimAUD 7.

Economou, A., Tzu-Chieh (Kurt), H., Heather, L., and James P. 2021. "Shape Machine: A Primer for Visual Computation." In A New Perspective of Cultural DNA, edited by Lee, J., 65–92. KAIST Research Series. Singapore: Springer.

Fish, J., and Stephen S. 1990. "Amplifying the Mind's Eye: Sketching and Visual Cognition." Leonardo 23 (1): 117–26.

Goodfellow, I., Jean P., Mehdi M., Bing, X., David, W., Sherjil, O., Aaron, C., and Yoshua, B. 2014. "Generative Adversarial Nets." In Advances in Neural Information Processing Systems. Vol. 27. Curran Associates, Inc.

G. Xi, Z. Wang, Q. Yang, W. Lv, X. Liu, Q.Wu, and J. Huang. 2020. "GAN-Based Virtual-to-Real Image Translation for Urban Scene Semantic Segmentation." Neurocomputing 394 (June): 127–35.

Huang, W., Hao, Z. 2018. "Architectural Drawings Recognition and Generation through Machine Learning." In ACADIA // 2018: Recalibration. On Imprecisionand Infidelity. [Proceedings of the 38th Annual Conference of the Association for Computer Aided Design in Architecture (ACADIA) ISBN 978-0-692-17729-7] Mexico City, Mexico 18-20 October, 2018, Pp. 156-165. CUMINCAD.

Ismail, A. S., Kherun N. A., and Noorminshah A. I. 2017. "A Review on BIM-Based Automated Code Compliance Checking System." In 2017 International Conference on Research and Innovation in Information Systems (ICRIIS), 1–6.

Isola, P., J. Zhu, T. Zhou, and Alexei A. E. 2018. "Image-to-Image Translation with Conditional Adversarial Networks." ArXiv:1611.07004 [Cs], November.

Lorenz, C., Bleil, D. S. 2019. "Machine Learning in Design Exploration: An Investigation of the Sensitivities of ANN-Based Daylight Predictions." In Ji-Hyun Lee (Eds.) "Hello, Culture!" [18th International Conference, CAAD Futures 2019, Proceedings / ISBN 978-89-89453-05-5] Daejeon, Korea, Pp. 75-87. CUMINCAD.

Mokhtar, S., Aleksandra, S., and Carlos, C. D. 2020. "Conditional Generative Adversarial Networks for Pedestrian Wind Flow Approximation." SimAUD 8.

Pranovich, S. 2004. "STRUCTURAL SKETCHER: A TOOL FOR SUPPORTING ARCHITECTS IN EARLY STAGES." In PhD Thesis, Technische Universiteit Eindhoven, The Netherlands. CUMINCAD.

Qian, W., Yang, X., and Hui, L. 2022. "A Self-Sparse Generative Adversarial Network for Autonomous Early-Stage Design of Architectural Sketches." Computer-Aided Civil and Infrastructure Engineering 37 (5): 612–28.

Rounce, G. 1998. "Quality, Waste and Cost Considerations in Architectural Building Design Management." International Journal of Project Management 16 (2): 123–27.

AUTHOR BIOGRAPHIES

HAN TU is a Master of Science candidate in the Design & Computation Group at MIT. Her research focuses on data-driven design workflows at both the urban and architectural scales. She holds a Diploma in Architecture from Tongji University and a Master of Architecture and Big Data from Tsinghua University. Her email address is hantu@mit.edu.

YICHAO SHI is a Master of Science candidate studying design computation at Georgia Tech. His research expertise lies in the application of generative algorithms to architectural and structural designs. He earned a Bachelor's degree in Architectural Engineering from Tongji University and a Master's degree in Architecture from the Bartlett School of Architecture at UCL. His email address is yshi431@ gatech.edu.

MENG XU holds a Bachelor's degree in Architecture from Hunan University and a Master of Science degree in Urban Studies and Planning Program from Aalto University. Her research focuses on computational design methods for sustainable cities. She Her email address is meng.xu@aalto.fi.