DOI: https://doi.org/10.38027/ICCAUA2024EN0263

Automatic Design Line Detection in Urban Building Images through DCT

* [1] Ph.D. Candidate **Lina Davud** [2] Assoc. Prof. Dr. **Selin Yildiz**
[1 & 2] *Department of Architecture, Faculty Of Architecture, Yildiz Technical University, Türkiye*
E-mail [1]: lina.alkhodari@std.yildiz.edu.tr , E-mail [2]: semir@yildiz.edu.tr

Abstract
The conversion of urban building images into design lines is a valuable technique used in architecture. It aids in the analysis of design styles and building components. Recent studies have shown a significant interest in using computational methods to examine the design lines in architectural images, such as plans, facades, and streetscapes. This research presents an automated approach that utilizes kernel filters to extract edges and reduce noise from various architectural images. Additionally, it introduces a new filter that separates lines based on their directions utilizing the Discrete Cosine Transform (DCT). The proposed technique generates distinct images that display the vertical, horizontal, and curved lines extracted from the original images. The results demonstrate that the proposed technique is efficient not only with architectural drawings but also with photographs of existing buildings. This technique opens the door for further experimentation in artificial intelligence, computational aesthetics, image rectification and calibration, and 3D building reconstruction.
Keywords: Design Lines; Urban Buildings; Computational Methods; Kernel Filters; Discrete Cosine Transform (DCT); Edge Extraction; Architectural Image Analysis.

1. Introduction

Recently, automation in the field of architecture has been widely utilized for various research and design tasks. Some of these tasks require the identification of building edges and design lines from urban images. Computer vision, a branch of image-based computer science, employs pixel values to deduce image content and develops mathematical techniques to retrieve 3D shapes from pictures (LeCun, Bengio, & Hinton, 2015). This crucial area of research enhances the efficiency of image examination by utilizing image features like color, shape, and texture to derive image content. Consequently, the extraction of building edges from urban images falls under the category of image feature extraction. In this study, the use of Laplacian kernel filters for edge detection from urban images is considered. Furthermore, this research explores the separation of lines into groups based on their direction utilizing Discrete Cosine Transform (DCT) and applies it to architecture images. The extracted building edges and design lines are subsequently employed in various computational methods, including image rectification and calibration, computational aesthetics application to existing images, augmented reality application design, automatic 3D building reconstruction, building recognition and classification, and other studies related to building facades, such as design generation and style analysis.

2. Related Works

Various researchers have conducted studies on the use of building edges and design lines. For instance, Duan W. and Allinson N. have proposed an automated approach to rectify facade images by estimating vanishing points and grouping building lines. They utilized the Canny edge detector to identify the edges of the building images that require rectification (Shan & Zhang, 2022). Additionally, the outline of buildings has been employed in augmented reality (AR) applications, where computer-generated elements are integrated into real space. Orhei et al. have presented a novel filter operator for detecting building edges, which enhances the extraction of facade features for AR applications (Ciprian, Silviu, & Radu, 2020). Similarly, Ran W. et al. have developed a CNN and transformer-based model to automatically extract building outlines from high-resolution aerial images. This approach proves useful for tasks like change detection and disaster assessment, as stated by the researchers (Ran, Yuan, Shi, Fan, & Shibasaki, 2023). Line detection in architectural facades is a significant area of research in the field of architecture. Detecting lines in urban images has practical implications for reconstruction, recognition, navigation, and scene understanding Aziz Amen, 2017; Aziz Amen & Nia, 2018; Ho et al., 2023). To address this, G. Wan and S. Li have proposed an automated method to segment facades using detected lines and vanishing points. Initially, the lines in the image are detected and adjusted based on the y-direction of the image, allowing for the identification of distinct segments representing the building facades (Wan & Li , 2011).

Various researchers have conducted studies on the use of building edges and design lines. For instance, Duan W. and Allinson N.have proposed an automated approach to rectify facade images by estimating vanishing points and grouping building lines. They utilized the Canny edge detector to identify the edges of the building images that require rectification (Duan & Allinson, 2009). Additionally, the outline of buildings has been employed in augmented reality (AR) applications, where computer-generated elements are integrated into real space. Orhei et al. have presented a novel filter operator for detecting building edges, which enhances the extraction of facade features for AR applications (Ciprian, Silviu, & Radu, 2020). Similarly, Ran W. et al. have developed a CNN and transformer-based model to automatically extract building outlines from high-resolution aerial images. This approach proves useful for tasks like change detection and disaster assessment, as stated by the researchers (Ran, Yuan, Shi, Fan, & Shibasaki, 2023). Line detection in architectural facades is a significant area of research in the field of architecture. Detecting lines in urban images has practical implications for reconstruction, recognition, navigation, and scene understanding. To address this, G. Wan and S. Li have proposed an automated method to segment facades using detected lines and vanishing points. Initially, the lines in the image are detected and adjusted based on the y-direction of the image, allowing for the identification of distinct segments representing the building facades (Wan & Li , 2011).

In addition, the design lines of architectural structures, including their direction, numbers, and metric values, play a crucial role in topics such as computational aesthetics, facades, and street-view analysis (Sardenberg & Becker, 2022) and (Aydin & Mirzaei, 2021). To assess the visual quality of urban spaces, the various elements in the urban environment need to be defined and evaluated based on aesthetic factors. This analysis can be conducted using traditional measuring techniques or computational aesthetics methods. Computational aesthetics focuses on the development of computational approaches that can make aesthetic decisions similar to humans (Hoenig, 2005). There are numerous aesthetic variables that can be quantified for visual quality analysis, such as balance, symmetry, rhythm, proportion, and others. For some of these variables, it is necessary to detect the design lines of the architectural facade based on their direction and length. In the study conducted by Santosa & Fauziah, the vertical and horizontal lines of the facades analyzed in their research were manually identified for use in their computational aesthetics application. These lines were included in the interface aesthetics variables formula used in the study to measure balance and symmetry (Santosaa & Fauziah, 2017). This process can be automated when the design lines of the facade are automatically detected and categorized.

Additional factors that may be involved in identifying design lines in urban images pertain to computer-based aspects. These aspects focus on the extraction and detection of features from images, whether they are high-level or low-level features. This aids in various applications, such as automatically classifying buildings based on their style (Duan & Allinson, 2009). This classification is determined by the prominent line style of the building's components, whether they are vertical, horizontal, or curvaceous. Consequently, the categorization of lines based on their orientation in different images is deemed advantageous. Moreover, this method proposed in our study can also be utilized for automatic 3D building reconstruction (Dolotov, 2014) and the automated analysis of facades and street views in relation to the dominant line style (Esmaili, Charehjoo, & Hoorijani, 2020), edges, and heights (Teeravech, Nagai, Honda, & Dailey, 2014).

3. Preliminary

To achieve the objective of this study, various techniques were examined and assessed for potential application. This section delineates the exploration and presentation of these techniques.

3.1 Laplacian Operator

The Laplacian is a two-dimensional isotropic calculation of the second spatial derivative of an image. Its purpose is to identify areas within the image where there is a sudden and significant change in intensity (Jing, Liu, Wang, Zhang, & Sun, 2022). As a result, it is commonly employed in the process of detecting edges in images (Idris, Abdullah, Abdul Halim, & Selimun, 2022). The Laplacian operator, denoted as $L_{x,y}$, can be obtained by applying a convolution filter to an image represented by pixel intensities $I_{x,y}$. Convolution is an essential mathematical operation that holds significant importance in a wide range of image processing operators (Waheed, Deng, & Liu, 2020). This operation involves the multiplication of two arrays containing numbers, which may have different sizes but possess the same dimensionality. As a result of the convolution process, a third array of numbers with identical dimensions is generated. This method is commonly utilized in image processing to develop operators that calculate output pixel values based on linear combinations of particular input pixel values. In image processing, one input array typically represents a grayscale image, while the second array, referred to as the kernel, is usually smaller and has a two-dimensional structure (though it could also be a single-pixel width). To exemplify the convolution concept, Figure 1 presents an image and kernel, serving as visual aids (Zou, 2023).

$I_{1,1}$	$I_{1,2}$	$I_{1,3}$	$I_{1,4}$	$I_{1,5}$	$I_{1,6}$	$I_{1,7}$	$I_{1,8}$
$I_{2,1}$	$I_{2,2}$	$I_{2,3}$	$I_{2,4}$	$I_{2,5}$	$I_{2,6}$	$I_{2,7}$	$I_{2,8}$
$I_{3,1}$	$I_{3,2}$	$I_{3,3}$	$I_{3,4}$	$I_{3,5}$	$I_{3,6}$	$I_{3,7}$	$I_{3,8}$
$I_{4,1}$	$I_{4,2}$	$I_{4,3}$	$I_{4,4}$	$I_{4,5}$	$I_{4,6}$	$I_{4,7}$	$I_{4,8}$
$I_{5,1}$	$I_{5,2}$	$I_{5,3}$	$I_{5,4}$	$I_{5,5}$	$I_{5,6}$	$I_{5,7}$	$I_{5,8}$
$I_{6,1}$	$I_{6,2}$	$I_{6,3}$	$I_{6,4}$	$I_{6,5}$	$I_{6,6}$	$I_{6,7}$	$I_{6,8}$

$k_{1,1}$	$k_{1,2}$	$k_{1,3}$
$k_{2,1}$	$k_{2,2}$	$k_{2,3}$
$k_{3,1}$	$k_{3,2}$	$k_{3,3}$

Figure 1. The convolution concept, the image is shown in the left, the kernel in the right.

The process of convolution is carried out by sliding the kernel over the image in a systematic manner, typically starting from the top left corner. This sliding motion ensures that the kernel covers every possible position within the boundaries of the image. It is important to note that different implementations may handle the edges of the image differently, which will be further explained below. At each position of the kernel, a corresponding output pixel is determined. This is done by multiplying the value of the kernel with the corresponding value of the underlying image pixel for each cell within the kernel. These multiplied values are then added together to obtain the final value of the output pixel. Using this approach, as an example, the value of $I_{5,7}$ pixel in the resulting image can be calculated as:

$$O_{5,7} = I_{4,6}\, K_{1,1} + I_{4,7}\, K_{1,2} + I_{4,8}\, K_{1,3} + I_{5,6}\, K_{2,1} + I_{5,7}\, K_{2,2} + I_{5,8}\, K_{2,3} + I_{6,6}\, K_{3,1} + I_{6,7}\, K_{3,2} + I_{6,8}\, K_{3,3}$$

If the image contains M rows and N columns, and the kernel consists of m rows and n columns, then the resulting image will have M - m + 1 rows, and N - n + 1 columns. This can be expressed mathematically as the convolution operation and the general formula is calculated as follows:

$$O_{(i,j)} = \sum_{k=1}^{m}\sum_{l=1}^{n} I(i+k-1, j+l-1)\, k(k,1)$$

where i runs from 1 to M - m + 1 and j runs from 1 to N - n + 1 (Fan, et al., 2022).

Convolution algorithms often result in larger output images when the kernel extends beyond the image boundaries. The implementation usually involves shifting the kernel so that its top left corner is aligned within the image, resulting in overlap with the right and bottom borders. This method has the advantage of preserving the input image size. However, it becomes necessary to assign values to pixels outside the image area in order to calculate the pixels at the bottom and right edges of the output image. A common approach is to assign a value of zero to these pixels, but this can lead to distortions in the output image. If one utilizes a convolution method following this approach, it is advisable to eliminate any areas that fall outside the boundaries of the image. To do this, simply remove (n-1) pixels from the right side and (m-1) pixels from the bottom of the image. Figure 2 shows different sizes of the Laplacian kernels. Each size of the Laplacian kernel corresponds to a specific level of edge detection efficiency (Kong, Akakin, & Sarma, 2013).

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}
\begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 2 & 1 & 0 \\ 1 & 2 & -16 & 2 & 1 \\ 0 & 1 & 2 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}
\begin{bmatrix} 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 3 & 3 & 8 & 1 & 0 \\ 1 & 3 & 0 & -7 & 0 & 3 & 1 \\ 1 & 3 & -7 & -24 & -7 & 3 & 1 \\ 1 & 3 & 0 & -7 & 0 & 3 & 1 \\ 0 & 1 & 3 & 3 & 3 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 \end{bmatrix}
\begin{bmatrix} 0 & 0 & 3 & 2 & 2 & 2 & 3 & 0 & 0 \\ 0 & 2 & 3 & 5 & 5 & 5 & 3 & 2 & 0 \\ 3 & 3 & 5 & 3 & 0 & 3 & 5 & 3 & 3 \\ 2 & 5 & 3 & -12 & -23 & -12 & 3 & 5 & 2 \\ 2 & 5 & 0 & -23 & -44 & -23 & 0 & 5 & 2 \\ 2 & 5 & 3 & -12 & -23 & -12 & 3 & 5 & 2 \\ 3 & 3 & 5 & 3 & 0 & 3 & 5 & 3 & 3 \\ 0 & 2 & 3 & 5 & 5 & 5 & 3 & 2 & 0 \\ 0 & 0 & 3 & 2 & 2 & 2 & 3 & 0 & 0 \end{bmatrix}$$

3x3 5x5 7x7 9x9

Figure 2. Different sizes of Laplacian filters.

3.2 Discrete Cosine Transform (DCT)

The Discrete Cosine Transform (DCT) is extensively used in the disciplines of signal processing and data compression. It provides a means of representing a function or signal using a combination of sinusoids with different frequencies and amplitudes. As a result, the DCT is a mathematical transformation that converts a real-valued signal, sampled in discrete time and amplitude, from the time domain to the frequency domain within a finite duration. The DCT transformation is a computation that generates a set of N real numbers, denoted as $X0$, ..., XN−1, from a given vector of N real numbers, denoted as $x0$, ..., xN−1, using the following formula (Hong, et al., 2023):

$$X_k = \sum_{n=0}^{N-1} x_n \cos\left[\frac{\pi}{N}\left(n+\frac{1}{2}\right)k\right]$$

For $k = 0, \ldots N-1$

Therefore, the 2-dimensional DCT (2D DCT) is essentially the one-dimensional DCT conducted sequentially along the rows and subsequently along the columns (or vice versa). In other words, the mathematical representation of the 2D DCT for an N1xN2 matrix can be described by the formula (Hong, et al., 2023):

$$X_{k1,k2} = \sum_{n_1=0}^{N_1-1}\sum_{n_2=0}^{N_2-1} x_{n_1,n_2} \cos\left[\frac{\pi}{N_1}\left(n_1+\frac{1}{2}\right)k_1\right] \cos\left[\frac{\pi}{N_2}\left(n_2+\frac{1}{2}\right)k_2\right]$$

For $k_i = 0, 1, 2, \ldots, N_i - 1$

The result of the DCT on an image yields a matrix that depicts the frequencies. These frequencies are arranged in a zig-zag pattern, from highest to lowest, when moving diagonally from the upper left corner to the bottom right corner. It is essential to acknowledge that the higher frequencies capture the overall features of the image, while the lower frequencies capture the finer details. Moreover, the frequencies located in the upper triangle, above the diagonal, represent the vertical edges that run from top to bottom. As we progress downwards, these frequencies gradually start to be influenced by the horizontal edges. On the contrary, the values found in the lower triangle, below the diagonal, correspond to the horizontal edges running from left to right. These frequencies are increasingly influenced by the vertical edges as we move towards the right. Figure 3 visually demonstrates this concept.

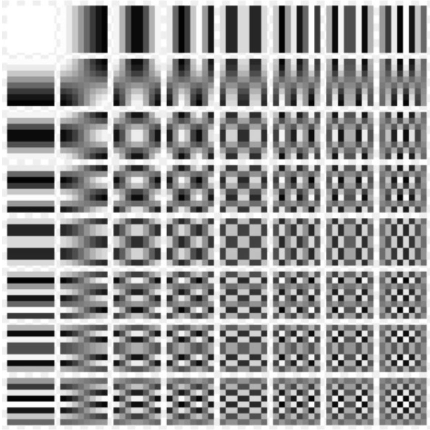


Figure 3. Logical visualization of the distribution of frequency directions in the spatial domain within the DCT matrix.

Most data processing applications that utilize DCT alter the data within the frequency domain due to the convenience of extracting content based on frequency rather than doing that in the spatial domain. Accordingly, any manipulation of the data entails implementing specified masks to zero out, amplify, or diminish values that fall within that mask in the frequency domain. This process enables the removal, enhancement, or reduction of the influence of said content in the original image within the spatial domain. Finally, the inverse of the 2D DCT is transforming back the image from the frequency domain to the spatial domain. It can be easily computed using the following formula (Hong, et al., 2023):

$$X_{n_1,n_2} = \sum_{k_1=0}^{N_1-1} \sum_{k_2=0}^{N_2-1} x_{k_1,k_2} \cos\left[\frac{\pi}{N_1}\left(n_1+\frac{1}{2}\right)k_1\right] \cos\left[\frac{\pi}{N_2}\left(n_2+\frac{1}{2}\right)k_2\right]$$

For $n_i = 0, 1, 2, \ldots, N_i - 1$

4. Proposed Method

In the process of identifying and separating design lines of urban façade images, the proposed methodology of this study involves an extensive experimentation that aims to determine and capture the various design lines based on their direction within the images. This experimentation is comprised of three main steps that are crucial for effectively obtaining desired results and improving the overall understanding of the urban façade design elements. These steps are visually represented in Figure 4, providing a clear visual guide for the entire process. The first step in this methodology is the data capturing phase, where a varied selection of urban images is carefully chosen from the renowned image platform "Pinterest". This specific selection is made using the search terms "elevation" and "architectural façade", ensuring that the chosen images are relevant and specifically focused on the desired subject matter. These images serve as the foundation for the subsequent steps, forming the basis for further analysis and design line extraction. Moving on to the second step, the data preprocessing phase plays a crucial role in preparing the selected images for analysis. There are five key procedures involved in this phase, each serving a specific purpose in enhancing the quality and optimizing the images for subsequent analysis. The first procedure involves a resizing process, where the dimensions of the images are adjusted to ensure consistency and uniformity. This involves modifying the width and height of the images to align with the smaller dimension, resulting in a standardized format that facilitates accurate analysis and comparison. Following the resizing procedure, the second step in data preprocessing entails converting the images from the RGB color space to greyscale. This conversion allows for a simplified and more focused analysis, as it eliminates any color variations and focuses solely on the visual elements presented in shades of grey. By eliminating color distractions, the subsequent analysis becomes more precise and effective in identifying the design lines based on their direction. Once the images have been converted to greyscale, the third procedure in data preprocessing involves identifying the edges within the images. This is achieved through the implementation of a 5 × 5 Laplacian kernel filter, which effectively detects and outlines the edges present in the images. By identifying these edges, the subsequent analysis can accurately pinpoint the design lines and distinguish them from the rest of the image components. This step plays a critical role in the overall process, as it serves as a foundation for subsequent analysis and line extraction. The fourth step involves performing a color inversion on the resulting image, which comprises a black background with white lines indicating the outlines of the building in the picture. This is implemented to highlight the edges present in the image. Finally, the fifth procedure in data preprocessing focuses on quantization. This involves converting the images into a binary representation, where each pixel is classified as either black or white. This binary representation is achieved by applying a threshold of 50%, where pixels with intensities below this threshold are classified as black, while those above are classified as white. This step simplifies the subsequent analysis and allows for a clearer identification of the design lines based on their direction. By reducing the image to simple black-and-white image, the subsequent analysis can more effectively identify and extract the desired design lines. Moving on to the final stage of the methodology, the altered images undergo the Discrete Cosine Transform (DCT) filter. This transformative filter converts the images from the spatial domain to the frequency domain, providing deeper insights into the design lines and their characteristics. However, it's important to note that the DCT filter used in this stage is specifically designed to cater to direction-based analysis. This means that it selectively excludes vertical and horizontal lines individually, allowing for a more focused and accurate extraction of the desired design lines as shown in table 1. This directional filtration ensures that only the relevant design lines, based on their direction, are captured and analyzed in the frequency domain. In light of this, three dynamic masks were implemented. Each mask serves the purpose of excluding lines in a specific direction. The masks and their objectives are depicted in Figure 5. Upon the application of the desired masks, the image undergoes a process whereby it is transformed back to its spatial domain through the utilization of the inverse of the 2D DCT.

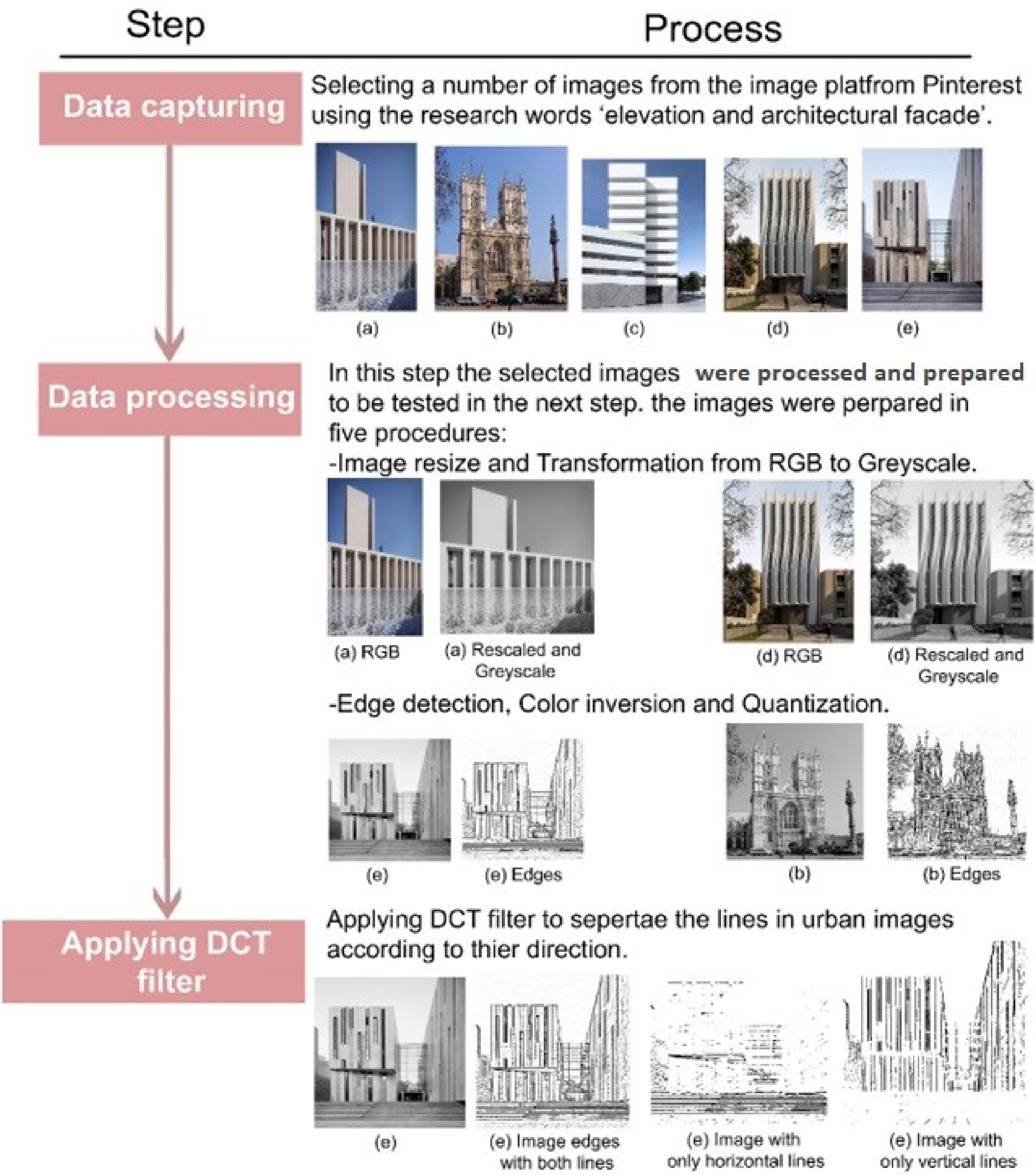
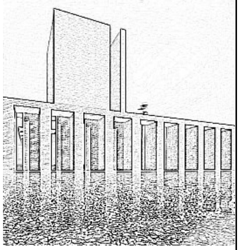
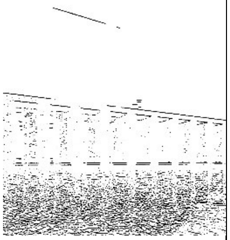
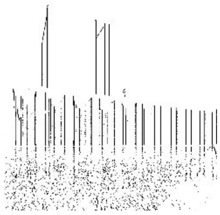
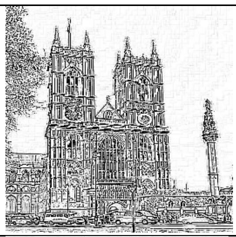
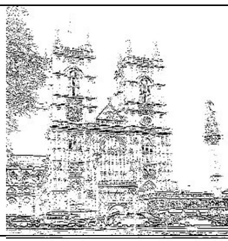
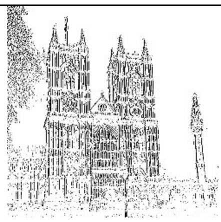
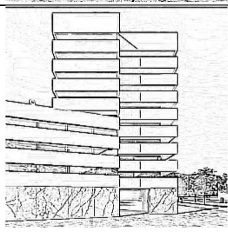
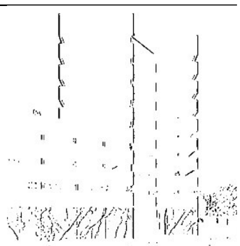
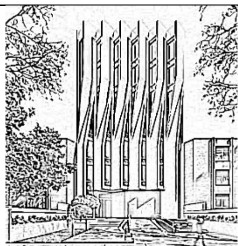
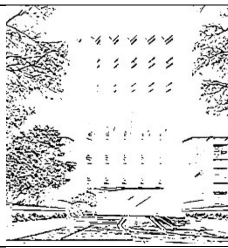
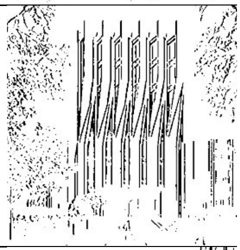
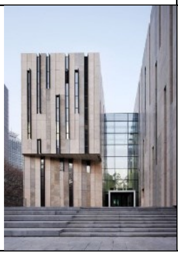
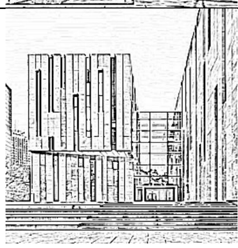
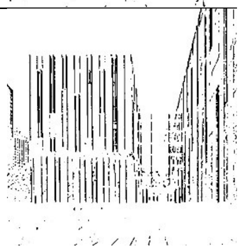


Figure 4. The proposed methodology of this study (Developed by Author).

Table 1. The images used in the experimentation and the obtained results.

Study case	RGB image	Image rescaled and Greyscale	Image edge detected	Horizontal lines only (vertical removed)	Vertical lines only (horizontal removed)
(a)					
(b)					
(c)					
(d)					
(e)					

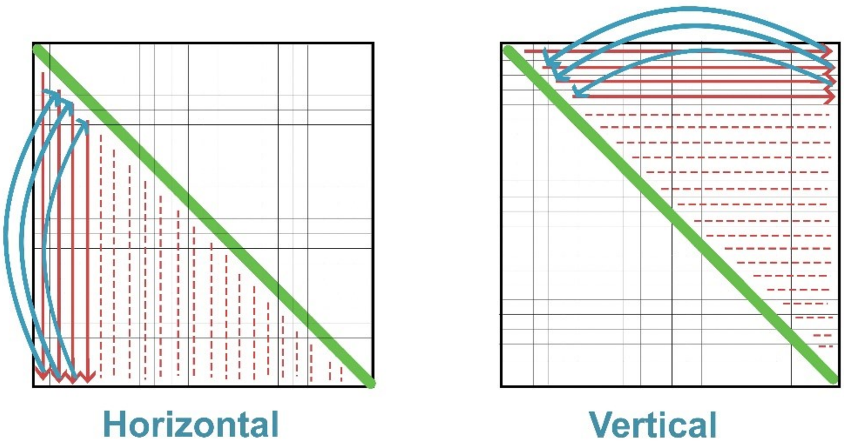


Figure 5. Masks utilized for the removal of horizontal (left) and/or vertical (right) edges within the Discrete DCT matrix (Developed by the author)

5. Results and Discussions

Automation has made its way into various domains, including architectural design. The ongoing advancements in technology present numerous opportunities to streamline human tasks by executing them with precision and efficiency. In this study, a method is proposed for determining and capturing the design lines of urban images of architectural façade. For this purpose, an experiment was conducted on a set of selected images. These images were preprocessed in different steps, among these steps the edges of the images were detected using a 5x5 Laplacian kernel filter. Following the preprocessing of the chosen images, the DCT filter was employed to separately capture the horizontal and vertical lines within the images. This suggested technique for automatic identification of design lines is regarded as advantageous for numerous fields that deal with the analysis, reconstruction, rectification, and other objectives related to architectural imagery. For future works, it is recommended to find an algorithm that can automatically calculate the lengths of these separated design lines. This step is helpful for the studies concerned with finding dominant line in architectural oeuvres and styles.

6. Conclusions

Technology and automation have become essential aspects of architectural design in recent years. Either fully or semi-automated processes, technology has been showing growing interest from researchers in different domains including architectural design. This interest stems from the numerous advantages these technologies offer, such as time efficiency, precise outcomes, ease of modifications, and enhanced levels of creativity, which are crucial elements in the design process. The subjects of computer vision and image processing are of great importance in research, as they play a crucial role in various processes involving images, such as generating images, classifying them, and conducting different forms of analysis. This study presents an automated approach for detecting design lines. The method derived from this study has the potential to serve as an effective initial stage for subsequent processes involving architectural images. Given that the conventional approach for identifying and capturing design lines is typically done manually, the introduction of this automated method is deemed advantageous for this task.

Acknowledgements

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Conflict of Interests

The Author(s) declare(s) that there is no conflict of interest.

References

Aydin, Y., & Mirzaei, P. (2021). novel mathematical model to measure individuals' perception of the symmetry level of building facades. *Architectural Engineering and Design Management, 18*, 1-18. doi:10.1080/17452007.2020.1862042

Aziz Amen, M. (2017). The inspiration of Bauhaus principles on the modern housing in Cyprus. Journal of Contemporary Urban Affairs, 1(2), 21–32. https://doi.org/10.25034/ijcua.2017.3645

Aziz Amen, M., & Nia, H. A. (2018). The dichotomy of society and urban space configuration in producing the semiotic structure of the modernism urban fabric. Semiotica, 2018(222). https://doi.org/10.1515/sem-2016-0141

Ho, T. P., Stevenson, M. M., & Thompson, J. J. (2023). Perceived Urban Design Across Urban Typologies in Hanoi. Journal of Contemporary Urban Affairs, 7(2), 156-170–156–170. https://doi.org/10.25034/IJCUA.2023.V7N2-10

Ciprian, O., Silviu, V., & Radu, V. (2020). A Novel Edge Detection Operator for Identifying Buildings in Augmented Reality Applications. *Communications in Computer and Information Science.* doi: https://doi.org/10.1007/978-3-030-59506-7_18

Dolotov, A. (2014). *Semi-automatic Modeling of Facades based on image references.* Master thesis, University of Dublin, Trinity College.

Duan, W., & Allinson, N. (2009). Automatic Approach for Rectifying Building Facades from a Single Uncalibrated Image. *In Proceedings of the 6th International Conference on Informatics in Control, Automation and Robotics - Robotics and Automation*, (s. 37-43). doi:10.5220/0002191600370043

Esmaili, F., Charehjoo, F., & Hoorijani, N. (2020). Analyzing and Evaluating Facades with a Special Approach to Visual Aesthetics Using the Grid Method (Case Study: Enqelab Street in Sanandaj). *Bagh-E Nazar, 17*. doi:10.22034/bagh.2019.165147.3934

Fan, L., Zhao, Z., Wang, K., Dutt, A., Wang, J., Buddhiraju, S., . . . Fan, S. (2022). Multidimensional convolution operation with synthetic frequency dimensions in photonics. *Physical Review Applied, 18*, 034088. doi:10.1103/PhysRevApplied.18.034088

Hoenig, F. (2005). Defining computational aesthetics. *Computational Aesthetics in Graphics, Visualization and Imaging (2005)*, (s. 13-18).

Hong, Q., He, B., Zhang, Z., Xiao, P., Du, S., & Zhang, J. (2023). Circuit Design and Application of Discrete Cosine Transform Based on Memristor. *IEEE Journal on Emerging and Selected Topics in Circuits and Systems, 13*(2), 502-513. doi:10.1109/JETCAS.2023.3243569

Idris, B., Abdullah, L., Abdul Halim, A., & Selimun, M. (2022). Comparison of edge detection algorithms for texture analysis on copy-move forgery detection images. *International Journal of Advanced Computer Science and Applications, 13*, 10.

Jing, J., Liu, S., Wang, G., Zhang, W., & Sun, C. (2022). Recent advances on image edge detection: A comprehensive review. *Neurocomputing*, 259-271. doi:10.1016/j.neucom.2022.06.083

Karanwal, S. (2021). *International Journal of Engineering and Manufacturing (IJEM), 11*, 1-10. doi:10.5815/ijem.2021.01.01

Kong, H., Akakin, H., & Sarma, S. (2013). A Generalized Laplacian of Gaussian Filter for Blob Detection and Its Applications. *IEEE Transactions on Cybernetics, 43*(6), 1719-1733. doi:10.1109/TSMCB.2012.2228639

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep Learning. *Nature, 521*. doi:10.1038/nature14539

Ran, W., Yuan, W., Shi, X., Fan, Z., & Shibasaki, R. (2023). Hybrid Feature Embedding for Automatic Building Outline Extraction. *In IGARSS 2023-2023 IEEE International Geoscience and Remote Sensing Symposium* (s. 3652-3655). IEEE.

Santosaa, H., & Fauziah, N. (2017). Aesthetic evaluation of restaurants facade through public preferences and computational aesthetic approach. *IPTEK Journal of Proceedings Series*. doi:10.12962/j23546026.y2017i3.2434

Sardenberg, V., & Becker, M. (2022). Computational Quantitative Aesthetics Evaluation. *eCAADe 2022: Co-creating the Future - Inclusion in and through Design*, (s. 567-574). doi:10.52842/conf.ecaade.2022.2.567

Shan, L., & Zhang, L. (2022). Application of Intelligent Technology in Facade Style Recognition of Harbin Modern Architecture. *Sustainability , 14*. doi:10.3390/su14127073

Teeravech, K., Nagai, M., Honda, K., & Dailey, M. (2014). Discovering repetitive patterns in facade images using a RANSAC-style algorithm. *ISPRS Journal of Photogrammetry and Remote Sensing, 92*, 38-53. doi:10.1016/j.isprsjprs.2014.02.018

Waheed, W., Deng, G., & Liu, B. (2020). Discrete Laplacian Operator and Its Applications in Signal Processing. *IEEE Access*, 89692-89707. doi:10.1109/ACCESS.2020.2993577

Wan, G., & Li , S. (2011). Automatic facades segmentation using detected lines and vanishing points. *4th International Congress on Image and Signal Processing*, (s. 1214-1217). doi:10.1109/CISP.2011.6100448

Zou, A. (2023). A survey of convolution. *Second International Conference on Statistics, Applied Mathematics, and Computing Science (CSAMCS 2022)*, *12597*, s. 966--972. Nanjing, China. doi:10.1117/12.2672707