RESEARCH ARTICLE OPEN ACCESS

Automated Hoarding Dimension Measurement: A Computer Vision-Based Approach for Accurate Billboard Size Estimation

Dr. Rozina Naz
Bachelor in Computer Application
Sadabai Raisoni Women's College
Nagpur, India

Ms. Nashrah Khan
Bachelor in Computer Application
Sadabai Raisoni Women's College
Nagpur, India

Mrs. Leena Deshmukh
Bachelor in Computer Application
Sadabai Raisoni Women's College
Nagpur, India

Ms. Reyina Jhamnani
Bachelor in Computer Application
Sadabai Raisoni Women's College
Nagpur, India

ABSTRACT

Measuring billboards and hoardings accurately is essential for ensuring regulatory compliance, tax assessments, and effective urban planning. Traditional methods often rely on manual measurements, which can be time-consuming, prone to errors, and risky for workers. To address these challenges, automated image processing offers a safer and more efficient alternative. This paper explores how computer vision techniques can be used to calculate hoarding dimensions with precision. By incorporating a standardized 2x2 meter reference board, the system establishes a reliable pixel-to-real-world scaling factor, enabling accurate measurements. Leveraging technologies such as OpenCV for image analysis and Flask for backend processing, this automated approach significantly reduces human effort while improving accuracy and safety. This advancement has the potential to streamline outdoor advertising management, benefiting advertisers, urban planners, and regulatory authorities by providing a scalable and modernized solution.

Keywords:— Image Processing, Computer Vision, Hoarding Measurement, OpenCV, Pixel-to-Real-World Scaling, Automated Measurement System, Outdoor Advertising.

I. INTRODUCTION

Traditionally, measuring hoardings and billboards has been a manual process, requiring significant time and human effort. These conventional methods often lack efficiency and come with considerable safety risks, especially when workers need to climb elevated structures for measurement. Additionally, manual approaches are prone to errors, leading to discrepancies that can impact regulatory compliance, tax calculations, and overall operational accuracy. In densely populated urban areas, where strict municipal guidelines govern hoarding dimensions, there is a growing need for a more precise, efficient, and safer alternative.

To overcome these challenges, the Hoarding Dimensions Calculator introduces an automated solution utilizing advanced image processing techniques. By incorporating a standardized 2x2 meter reference board as a calibration tool, the system employs computer vision algorithms to accurately measure hoarding dimensions from uploaded images. This eliminates the reliance on manual intervention, significantly reducing human effort, minimizing errors, and improving safety. The platform is designed as a scalable web application**,** seamlessly integrating into existing workflows while providing a user-friendly interface for a wide range of users.

The system's architecture is powered by Python and OpenCV for real-time image processing, with Flask handling server-side operations. This combination ensures an efficient, responsive, and capable platform for processing large datasets. Users can upload images either manually or via URLs, enabling quick and precise dimension extraction.

Beyond measurement accuracy, the system offers significant benefits in regulatory compliance by verifying hoarding dimensions against legal standards. Additionally, automation aids tax-related processes by providing reliable measurements for tax assessments. By streamlining these critical functions, the Hoarding Dimensions Calculator serves as a valuable tool for urban planners, advertisers, and regulatory bodies in outdoor advertising management.

The key contributions of this paper include:

- Automated Dimension Measurement: The system employs advanced image processing techniques to determine hoarding dimensions, eliminating the inefficiencies and inaccuracies associated with traditional manual measurement methods.
- Precision and Regulatory Compliance: By utilizing a standardized reference board, the system enhances measurement accuracy, ensuring compliance with established regulations and minimizing errors in hoarding size assessments.
- Improved Safety and Operational Efficiency: Automation reduces the need for physical measurement, lowering the risks associated with working at heights while streamlining the overall process.
- Enhanced Applications in Taxation and Urban Planning: The system facilitates accurate tax calculations and assists urban authorities in managing outdoor

advertisements more effectively through precise dimension verification.

II. Literature Review

The domain of computer vision and image processing has been significantly influenced by various landmark studies and comprehensive literature. Bradski and Kaehler [1] highlighted that "OpenCV has become a cornerstone library for building computer vision applications due to its open-source nature and extensive functionality." This versatility underpinned the detection and measurement tasks in this project.

Forsyth and Ponce [2] argued that "image processing must balance computational efficiency with accuracy, particularly in real-time applications." Their insights guided the optimization of algorithms used in detecting the red reference board.

Jain [3] emphasized, "Effective image processing begins with accurate pre-processing techniques to reduce noise and enhance critical features." This served as the foundation for applying contour detection and bounding box methods to isolate relevant regions within images.

Dalal and Triggs [6] observed, "Precise localization of features within an image is the key to enabling high-level recognition tasks." This perspective informed the design of the system for accurately identifying hoarding boundaries.

Szeliski [8] stated, "Bridging the gap between theory and implementation is critical for deploying computer vision solutions in the real world." This principle directly influenced the development of the user-friendly, scalable web application used in this project.

Python Software Foundation [4] described Python as "an ecosystem that supports rapid prototyping and integration of computationally intensive libraries, such as OpenCV and Flask." This characteristic made Python the ideal choice for building the software architecture of this project.

Flask Documentation [5] noted, "Flask's modularity allows developers to seamlessly integrate client-side and server-side functionalities," which facilitated the creation of the web-based interface for uploading and processing hoarding images.

Scikit-Image Documentation [9] emphasized, "Scikit-Image provides a robust collection of algorithms for efficient image processing," enabling the preprocessing of hoarding images before applying contour detection.

Gonzalez and Woods [10] remarked that "the evolution of digital image processing owes much to the effective representation and manipulation of image features." Their methodologies influenced the feature extraction techniques adopted in this project.

Goodfellow et al. [11] noted, "Deep learning models have revolutionized object detection and recognition tasks, enabling unprecedented accuracy." While traditional image processing techniques were used in this research, their observations provide a roadmap for integrating deep learning in future iterations.

Zhang [12] observed that "camera calibration and intrinsic parameter extraction play a pivotal role in translating pixel-based measurements into real-world dimensions." This insight was instrumental in the derivation of the pixel-to-foot scaling factor using the red reference board.

Sonka et al. [13] stated, "Image segmentation is a prerequisite for any successful image analysis pipeline, especially when working with heterogeneous environments." Their methods were foundational in isolating the red reference board from complex backgrounds.

Russell and Norvig [14] remarked, "AI techniques complement traditional algorithms by providing adaptive and intelligent problem-solving capabilities." This principle serves as a guiding light for future extensions of the project to incorporate AI-based enhancements.

III. System Design And Methodology

A. Reference Object for Scaling: The 2x2 Meter Red Board

The process begins with capturing an image of the hoarding, where a 2x2 meter red reference board is strategically placed in the same plane as the hoarding to serve as a standardized scale for precise dimensional analysis. This image is then uploaded through an HTML form, which sends the data to a backend server via a Flask-based API. Upon receiving the image, the backend initiates preprocessing by defining specific thresholds for the Blue, Green, and Red (BGR) color channels, as shown in Fig. 1. These thresholds, carefully calibrated using the OpenCV-Python library, are designed to identify and isolate pixels corresponding to the red reference board, separating them from the rest of the image. A binary mask is then generated, where pixels that fall within the defined range are marked as positive, while others are excluded as negative (Fig. 2a).

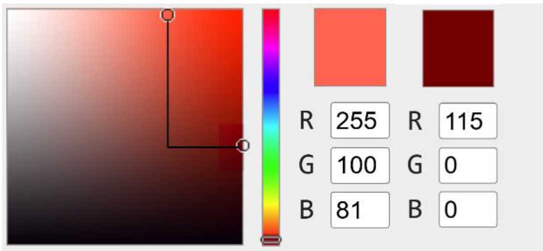

Figure 1: Defining the Color Range Illustration of the specified color range in BGR format used for detecting the red reference board in the image.

(a) (b)

Fig 2: (a) Masked image of with the positive pixels as white and negative masked as black. (b) Morphed image using erosion: Reducing the noise and smoothening the edges for easier detection.

Fig 3: The original image of the red board being processed and hence the red board being detected.

To enhance the detection's accuracy, the binary mask undergoes morphological processing, specifically an erosion operation (Fig. 2b), implemented using OpenCV's tools. During this step, a kernel defined as a 4×4 matrix of ones is applied to the binary mask. The erosion process effectively reduces noise and irrelevant regions by shrinking positive areas, thereby isolating the primary red region of interest. This refined mask ensures that only the most significant red region—representing the reference board—remains prominent, while eliminating false positives and small artifacts that could interfere with the analysis.

After refining the binary mask, the system leverages OpenCV's contour detection functionality to identify potential regions of interest. Detected contours are further processed using the approxPolyDP function, which simplifies contours into polygons. Each polygon is evaluated for convexity and area, ensuring only significant and relevant shapes are considered.

The next step involves verifying whether the identified contour corresponds to a square-like shape. This is achieved by analysing the contour's aspect ratio, which is calculated using the dimensions of the bounding rectangle. Only contours with an aspect ratio between 0.6 and 1.4 are retained, as this range aligns with the expected proportions of the red reference board. Additionally, contours must have an area exceeding 20 pixels to ensure relevance.

Once a valid contour is detected, the system draws a bounding rectangle around the identified region and annotates it with the message "Red Square Detected!" at the top of the bounding box (Fig. 3). After locating the largest square-like object, the process extracts the four corner points of the contour. These anchor points, combined with the known physical dimensions of the red reference board, are then used to compute the pixel-to-real-world scaling factor. This scaling factor plays a vital role in converting pixel-based measurements into accurate real-world dimensions. By harnessing OpenCV's robust image processing capabilities, the system ensures precise and consistent calculations for hoarding dimensions, forming a reliable foundation for subsequent analysis and measurement workflows.

B. Corner Selection For Hoarding Dimension Measurement

In the Hoarding Dimension Calculator, accurate corner selection is crucial for determining the exact dimensions of the hoarding. To achieve this, the system provides an interactive and intuitive user interface that enables users to manually position markers at the four corners of the hoarding in the uploaded image. This manual selection step is essential for converting pixel coordinates into real-world measurements and allows the system to calculate the hoarding's dimensions with precision.

The user's task is to identify and select the four corners of the hoarding by clicking on the image overlay. Each of the four corners is represented by a draggable circular point, which is visually indicated by a small green dot (Fig 4). The overlay, which is transparent, is precisely aligned with the dimensions of the uploaded image, giving the user a clear visual reference for accurate placement of the points. This alignment ensures that the markers can be placed with high precision.

As each of the four points is placed, the system immediately provides visual feedback by connecting the points with lines, thus forming the rectangular boundary of the hoarding. This real-time visualization helps the user confirm that the points are accurately placed and provides the option to adjust the placement if necessary. The immediate feedback ensures that users can verify their selections and adjust the markers in case of any inaccuracies, guaranteeing the correct delineation of the hoarding's area.

Once the points are placed, they remain adjustable, as they can be dragged to a new position as needed. This feature enables users to easily correct any initial placement errors, making the selection process more flexible. The ability to fine-tune the location of each marker is especially helpful in ensuring that the boundary of the hoarding is defined as accurately as possible.

The coordinates of the four selected corner points are then captured and stored in hidden form fields within the web interface. These coordinates, along with the annotated image containing the selected points, are submitted through an HTML form for backend processing. The backend uses this data to perform further calculations, including dimension

Fig 4: Draggable points being positioned at the corners of the hoarding, with the detected red reference box visible in the image near the hoarding.

extraction and scaling to real-world measurements. The captured data is integral to the accurate measurement and scaling process, forming the foundation for the system to provide reliable and precise dimensional analysis of the hoarding.

By allowing for accurate corner selection and providing immediate visual feedback, the system guarantees that the dimensions extracted from the image are both precise and dependable, laying the groundwork for the following automated dimension calculation process.

IV. Dimension Calculation

To achieve accurate real-world measurements, the system utilizes a pixel-to-foot scaling factor, derived from the known dimensions of the red reference board. This conversion is essential for translating the pixel measurements into the actual dimensions of the hoarding.

The width of the reference board in pixels is directly measured from the image. The Pixels Per Foot (PPF) is calculated by dividing the red board's width in pixels by its known real-world width in feet. The formula for calculating the PPF is as follows:

$$PPF = \frac{Red\ board's\ Width\ in\ pixels}{Known\ Width\ in\ feet}$$

Using this scaling factor, the system converts the hoarding's pixel dimensions into real-world measurements. The width and height of the hoarding in pixels (Hoarding's Width and Hoarding's Height) are converted to feet with the following formulas:

$$Hoarding's\ Width\ in\ feet = \frac{Hoarding's\ width\ in\ pixels}{PPF}$$

$$Hoarding's\ Height\ in\ feet = \frac{Hoarding's\ Height\ in\ Pixels}{PPF}$$

Through the use of OpenCV's image processing features, such as contour detection, the system accurately identifies the red reference board and calculates the pixel-to-foot scaling factor. This factor becomes the basis for all further dimension calculations.

This efficient method ensures that the hoarding's dimensions are calculated with precision, aligning pixel-based measurements with actual real-world specifications. The precise scaling guarantees compliance with standards and improves operational efficiency in hoarding management.

V. Conclusion And Future Scope

The Hoarding Dimensions Calculator marks a significant step forward in the automation of hoarding and billboard measurements. By incorporating advanced computer vision technologies into an intuitive web-based application, the system eliminates the inefficiencies, safety risks, and inaccuracies commonly associated with manual measurement techniques. The system's innovative use of a standardized red reference board and pixel-to-real-world scaling ensures accurate dimension calculations, promoting regulatory compliance and optimizing tax assessments for urban outdoor advertisements. This approach not only lowers labour costs and reduces safety risks but also provides a scalable, efficient solution for stakeholders in advertising and urban planning.

The system's robust backend infrastructure, built with Python, OpenCV, and Flask, delivers a potent yet user-friendly platform for real-time image processing and data analysis. The ability to precisely calculate hoarding dimensions from images highlights the potential of image processing technologies in solving practical challenges faced in urban management and advertising.

The Hoarding Dimensions Calculator also offers substantial room for enhancement to address more intricate use cases and introduce additional functionalities. Key areas for future development include:

1. Processing Complex Hoarding Shapes: The system can be extended to accommodate hoardings with irregular shapes, such as those with curved surfaces or ones that taper at the top or bottom. By integrating advanced algorithms, the platform can handle such complex designs, broadening its applicability to a wider range of hoarding types.

2. Structural Analysis and Durability Assessment: Future versions of the system could include the ability to capture images of the hoarding's backside, focusing on its supporting metal framework. Using advanced image processing and structural analysis algorithms, the system could assess the wind resistance of the structure and evaluate the stability and durability of the heavy framework. This enhancement would provide crucial

insights into the structural integrity and safety of the hoarding.

With these planned improvements, the Hoarding Dimensions Calculator could evolve into a more comprehensive tool, offering expanded capabilities for advertisers, urban planners, and regulatory authorities.

Acknowledgment

The authors express their sincere gratitude to Sir Manish Kandrikar from Embedded Creations for invaluable support and guidance throughout the course of this work.

References

[1] G. Bradski and A. Kaehler, *Learning OpenCV: Computer Vision with the OpenCV Library*. Sebastopol, CA: O'Reilly Media, 2008.

[2] D. A. Forsyth and J. Ponce, *Computer Vision: A Modern Approach*, 2nd ed. Upper Saddle River, NJ: Pearson, 2011.

[3] A. K. Jain, *Fundamentals of Digital Image Processing*. Englewood Cliffs, NJ: Prentice Hall, 1989.

[4] Python Software Foundation, "Python Language Reference, version 3.x," available at: https://www.python.org/doc/, 2025.

[5] Flask Documentation, "Flask: The Python Microframework for Building Web Applications," available at: https://flask.palletsprojects.com/.

[6] N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2005, pp. 886–893.

[7] R. Muthukumar, "Geometric Transformations in OpenCV," unpublished.

[8] R. Szeliski, *Computer Vision: Algorithms and Applications*. London: Springer, 2010.

[9] Scikit-Image Documentation, "Image Processing in Python with scikit-image," available at: https://scikit-image.org/.

[10] S. Birchfield, *Image Processing and Analysis*. Pacific Grove, CA: Thomson Learning, 2008. *"Image processing involves a set of algorithms applied to enhance, analyze, and manipulate images to extract meaningful information."*

[11] K. R. Castleman, *Digital Image Processing*, 2nd ed. Upper Saddle River, NJ: Prentice Hall, 1996. *"Digital image processing is an essential technology in modern-day computer vision and graphics applications."*

[12] A. S. Glassner, *Principles of Digital Image Synthesis*, vol. 1. San Francisco: Morgan Kaufmann, 1995. *"Understanding the basics of image formation is crucial to building advanced computer vision systems."*

[13] OpenCV Documentation, "Image Processing Techniques in OpenCV," available at: https://docs.opencv.org/. *"OpenCV provides a suite of tools to perform edge detection, contour identification, and geometric transformations."*

[14] J. Redmon and A. Farhadi, "YOLO9000: Better, Faster, Stronger," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 7263–7271. *"YOLO9000 integrates real-time object detection with a strong emphasis on speed and efficiency."*

[15] R. Gonzalez and R. Woods, *Digital Image Processing Using MATLAB*, 2nd ed. Upper Saddle River, NJ: Pearson, 2009. *"MATLAB offers an interactive platform to explore image enhancement and analysis algorithms."*

[16] J. Canny, "A Computational Approach to Edge Detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-8, no. 6, pp. 679–698, Nov. 1986. *"The Canny edge detector remains one of the most popular algorithms for edge detection in image processing."*

[17] D. G. Lowe, "Object Recognition from Local Scale-Invariant Features," in *Proc. Int. Conf. Computer Vision (ICCV)*, 1999, pp. 1150–1157. *"SIFT features are effective in identifying objects under varying conditions of scale and rotation."*

[18] A. P. Pentland, "Recognition by Components: A Theory of Human Image Understanding," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-9, no. 6, pp. 541–556, Nov. 1987. *"Object recognition can often be simplified by reducing objects into recognizable components."*

[19] T. Ojala, M. Pietikainen, and D. Harwood, "A Comparative Study of Texture Measures with Classification Based on Featured Distributions," *Pattern Recognition*, vol. 29, no. 1, pp. 51–59, 1996. *"Texture analysis plays a crucial role in identifying patterns and features within images."*

[20] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, 3rd ed. Upper Saddle River, NJ: Pearson, 2010. *"AI-powered techniques allow for sophisticated image analysis and autonomous decision-making processes."*

[21] A. Vedaldi and K. Lenc, "MatConvNet: Convolutional Neural Networks for MATLAB," in *Proc. ACM Int. Conf. Multimedia*, 2015, pp. 689–692. *"MatConvNet is a powerful deep learning framework designed for efficient processing of multimedia data."*

[22] TensorFlow Documentation, "TensorFlow: Scalable Machine Learning on Heterogeneous Distributed Systems," available at: https://www.tensorflow.org/. *"TensorFlow enables the development of scalable deep learning applications, including those in computer vision."*

[23] P. Viola and M. J. Jones, "Robust Real-Time Face Detection," *Int. J. Computer Vision*, vol. 57, no. 2, pp. 137–154, 2004. *"Robust real-time detection algorithms often rely on simple yet effective cascade classifiers."*

[24] M. Turk and A. Pentland, "Eigenfaces for Recognition," *J. Cognitive Neuroscience*, vol. 3, no. 1, pp. 71–86, 1991. *"Face recognition systems heavily benefit from the use of eigenfaces for feature extraction and classification."*