Journal of Theoretical and Applied Information Technology
30th June 2023. Vol.101. No 12

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

ESTIMATING THE LOCATION OF AN INDOOR CHAIR OBJECT FROM A SINGLE IMAGE USING A PERSPECTIVE GRID APPROACH

DARMA RUSJDI[1], YAYA HERYADI[2], GEDE PUTRA KUSUMA[3], EDI ABDURACHMAN[4]

[1,2,4]Computer Science Departement, BINUS Graduate Program - Doctor of Computer Science, Bina Nusantara University, Jakarta, Indonesia, 11480

[3]Computer Science Departement, BINUS Graduate Program - Master of Computer Science, Bina Nusantara University, Jakarta, Indonesia, 11480

[4]Trisakti Institute of Transportation and Logistic

E-mail: [1]darma.rusjdi@binus.ac.id, [2]yayaheryadi@binus.edu, [3]inegara@binus.edu, [4]ediabdurachman@gmail.com, edia@itltrisakti.ac.id

ABSTRACT

Estimating the 3D location of an object from a single camera is an important topic in the field of computer vision and computer graphics. The problem of transforming a 2D point from a single image to a 3D point is difficult unless it is located in the same plane, using an RGBD camera or at least using two images. Our study aims to discuss the perspective grid concept for estimating the 3D location of chair objects accurately on the floor surface using a single image from a cellphone camera. Our proposal to predict the 3D location of a single image from a camera goes through three experimental stages. First, setting the nine locations and four actual object orientations over the floor pattern in the room to get the bounding box position by utilizing the object detection pre-trained model. The second stage is the development of the perspective grid algorithm as a transformation of 2D points in the projected image to 3D points on the floor plane. The third stage is predicting the optimal object location in the image from the lower left and lower right bounding box positions (assumed to be on the floor surface where the z value is 0) with a perspective grid approach, to be projected into a 3D location prediction value on the floor plane. Then we evaluate the calculation of the deviation of the average prediction error from nine actual object locations, each object location, and each object orientation. The average error deviation result is 6.47 centimeters. This shows that the results are quite accurate compared to the dimensions of the object and the area of the room.

Keywords: *Bounding Box, Location Estimation, Object Detection, Perspective Grid, Single Image.*

1. INTRODUCTION

Drawing using CAD applications not only make drawings concise, and beautiful scenery, but also easy to modify, and high reuse rate. If designers are proficient in CAD applications, they can increase their productivity when designing and drawing interior spaces.[1][2] Layout design activity on interior work, and architecture with CAD models of 2D and 3D objects that are defined according to object categories. Object models are generally rigid, stored in an image library in a large number of types and categories, and have a large storage capacity. The CAD model is a synthesis object that is made according to real objects, especially from the manufacturer. Current CAD applications have a rendering function with lighting techniques and textured surfaces that can produce natural visualizations. Utilizing CAD models both 2D and 3D provides convenience in automatic drawing [3] especially to get the room dimensions and layout settings that contain the various categories of objects needed. Placement of each CAD object model into a plane or drawing space is very easy by calling the name of the required category object and then placing the location on a defined surface area followed by the orientation and scale of the object.

Previous studies related to object detection for location estimation resulted in the classification and localization of category objects in image scenes. The cost of consumer-grade color and depth (RGB-D) cameras, for example, the Microsoft Kinect, is widely used to reconstruct 3D indoor scenes at low cost [4]. Pose estimation is a complex job, it is easier to predict 6D poses from images produced by

RGBD cameras than from RGB cameras because 6D poses are a complex combination of orientation estimation in the form of 3D rotation of an object (raw, pitch, yaw) and estimation of 3D coordinate locations (X, Y, Z) [5].

Estimating the location of objects from a single image is still a trend of study and application in the field of computer vision and computer graphics in the last decade.[6]. Location estimation can be achieved using more than one image, tend to use cameras with additional depth values, or stereo cameras. But until now 3D perception can be achieved with just one image accurately to be something interesting.

The use of one camera is still a major problem in this study, namely the accuracy of predicting the location of 3D objects from a single image. Izadinia's approach to positioning and scaling objects in a room is iteratively optimized to match the input photo to the rendered scene, the trained image comparison metrics are used through a deep convolutional neural net. The study was carried out through 3d scene generation to match the location and orientation of objects.[7] [8] It does not yet show a 2D transformation to a 3D coordinate location.

2D to 3D transformation is the process of accurately transforming 2D image coordinates into 3D coordinates in real space. Unlike the pinhole camera process, a perspective projection process from 3D points in space becomes 2D points in the image. One of the requirements for finding a 3D location using a single camera is that the size of the object in the image that needs to be estimated must be known or the object is on the same surface.

Objects in the image may have different sizes when the object's distance from the camera changes, including the object's orientation changes. To understand the location of objects, it's easier to estimate the 3D location of a ball. Because with different orientations from the same distance, they would theoretically have the same size.

In the field of computer science and computer graphics, the problems above are technically related to the problem of 3D reconstruction based on CAD models from a single image [9], search or object recognition related to the classification or detection of objects in the image through a bounding box, segmentation so that the location of the object in the image is known. The use of one camera with a deep learning approach has been a trend of study since the last decade.

Another problem with using the deep learning method is that it requires a dataset with a large enough amount of data for training – testing and requires speed performance and memory capacity, in some cases using augmented techniques to overcome data shortages. To overcome this, we can use a pre-trained object detection model and utilize the position of the bounding box on objects that are recognized in the image.

The objective of this research is to predict the location of objects, we propose to apply the perspective grid method, which is usually used by painters or engineering designers, to visually embody ideas in the form of models.

In this research, we will use the best method from several pre-trained object detection models. Using an object detection model so that category objects can be identified and the location of the bounding box in the image can be identified. The use of the pre-trained model is an efficient step but its effectiveness is chosen from several pre-trained models that tend to be used by various studies.

Our research mainly provides novelty in the form of combining the use of the CNN method and the perspective grid model approach. The CNN method is used for object recognition and 2D location estimation in the form of bounding box locations from a single image, then followed by a perspective grid approach to estimate object locations on the floor surface based on bounding box location predictions. Our contribution is an algorithm to calculate the transformation from one image's 2D coordinates to the 3D location coordinates ($z = 0$) of a chair object on the floor.

2. RELATED WORK

2.1. Object Detection

Object detection is an application in computer vision that is most common in all walks of life to use, such as driverless car applications that can detect roads, objects in front of blocking vehicles and traffic signs, to count the number of people in a crowd or the number of ends stacked iron in a round or square shape. Other applications include text detection, face detection, pedestrian detection and remote sensing target detection. Object detection combines classification and localization. The input is an image that contains one or more objects. The output is the result of predicting the location of the object with the bounding box and the classification of objects from each bounding box.

According to Zou and friends,[10], object detection has gone through two historical periods, the first before 2014 is called the traditional object detection period, namely: HOG Detectors, Viola Jones Detectors, and Deformable Part-based Model

(DPM). Several important data sets and benchmarks have been used in the last two decades, including the utilization of MS-COCO, PASCAL VOC, ImageNet, etc. Second, after 2014 it is called the detection period based on deep learning. There are two types of CNN-based methods: first, CNN-based two-stage detectors: RCNN, Fast-RCNN, Faster-RCNN, Spatial Pyramid Collection Network (SPPNet), Feature Pyramid Network (FPN); and second, CNN-based single-stage detectors: You See Only Once (YOLO), Single Shot MultiBox Detector (SSD), RetinaNet, CenterNet. These models have previously been trained and used as benchmarks to refine existing models or test models.

2.2. Camera Calibration for Perspective Transformation.

Camera calibration required to capture geometric images is the first step for many computer vision and graphics tasks, from 3D reconstruction to image metrology to photographic editing.[11] The use of imagery for 3D location determination has been carried out based on camera calibration[12][13], Lopez uses a Deep image calibration camera which can recover from extrinsic (tilt, roll) and intrinsic parameters (focal length and radial distortion) problems from a single image [14] such as barrel distortion and pincushion distortion. (figure 1).[15]

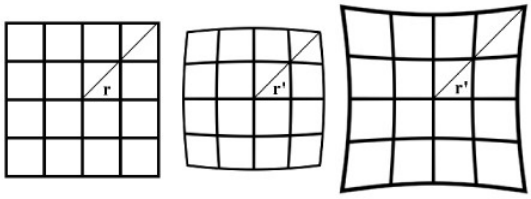


Figure 1. Radial distortion of a rectangular grid. Left, Middle, Right: No distortion. Barrel distortion. Pincushion distortion.

Park took the first step in the form of calibrating the camera under perspective projection [16] for intrinsic parameter calculation by calculating the relationship between the 3D world in the camera coordinate system and the 2D pixel coordinates in the image. Next uses the object segmentation algorithm to find the desired object, measures the size of the object, and after that uses the object size and the location of the object's center to estimate the 3D location using the intrinsic parameters obtained in the first step.

In another study, the perspective projection method was applied to the transformation of a 2D image plane into 3D coordinates and the estimation of a single image location was carried out through a direct linear transformation.[17][18][19]. Estimation of 6D poses from a single RGB image by first establishing a 2D-3D correspondence between the coordinates on the image plane and the object coordinate system, and then applying the variants of the PnP, homography, and RANSAC algorithms. [20][21]

The perspective projection [22] where reference distances are measured or assumed from features such as road markings or standard lane widths [23][24], combined with the 'vanishing points' where parallel lines meet in the image domain, provide parameters that allow camera calibration through algorithmic optimization [25].

Perspective projection and scale factor are handled by remotely mapping the corresponding image and real-world coordinates via homography (projection transformation).[26] Projective transformation is a technique of registering an image related to a reference image (fixed data) which is used to map a new location to the output image from the moving data pixel location (input image). Projective Transform is used to capture image angles and correct image distortions.[27]

The perspective transformation from 2D to 3D tends to use at least two images even from a single camera. The perspective grid approach on the image plane has not been used directly in the image to transform into 3D coordinates for one surface plane. Alberti's perspective grid is used as the basis for painting depictions and 3D designs [28] (see figure 2) in the fields of architecture, interiors, and engineering. The vanishing point of a single view image is used for the synthesis view and depth estimation [29], Composition sensitive photo capture application is used to detect dominant vanishing points in natural scenes.[30].

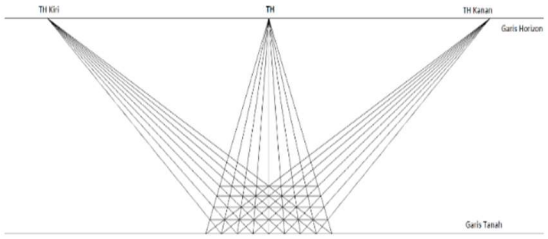


Figure 2. Alberti's Perspective Grid.

Currently, the research trend is to estimate the location through camera calibration using a reference medium in the form of a chessboard box. In general, it is used to estimate small objects and determine 3D shapes, but it is still not large enough to determine the size of furniture objects in the room.

Our approach develops a 2D coordinate transformation model from Alberti's perspective concept with 6 reference points on the floor plane. Our study aims to obtain a perspective grid transformation model with an accurate interpolation approach to the indoor floor field of a single image. The benefits of estimating the location of objects on the floor surface as a basis for placing CAD model objects in 3D depictions of interior/architecture fields as well as information on the location of objects for the robotic field.

3. METHODS

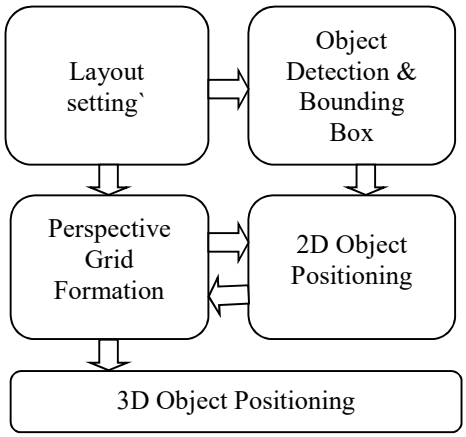


Figure 3. Research Step

Our proposal aims to produce accurate estimates of object locations in images that are above the floor surface from the direction of the camera position using a perspective grid-based object detection approach. In this study, we used two kinds of datasets, namely primary and secondary datasets. The primary dataset is generated by setting the location of a chair object in space above the floor surface. The secondary dataset is used for the object detection process.

The discussion of the methods we use is as follows: The first step is setting the layout and creating the input image of the chair object. The second is the detection of category objects in images using pre-trained models and available datasets (COCO and Pascal VOC) and for generating 2D bounding box position parameters. The third is the formation of a perspective grid coordinate system on the image based on 6 reference points which form three line equations to get one vanishing point. The fourth determines the 2D location of the chair object in a single image used for transformation from 2D to 3D position on the floor surface and the fifth determines the 3D location on the floor surface, where the x value uses the comparison equation of 2 congruent triangles, the y value through Newton's interpolation equation, and the z value has a value of 0 from the position representation of the bottom left and bottom right of the Bounding box. The object location is a point based on the 3D CAD model from the camera direction.

3.1. Layout setting.

The aim is to get the coordinates of the object's location in the photo image by utilizing the bounding box values of the lower left corner and lower right corner which indicates the location above the floor surface. Currently, there are 36 labeled images available (chair and bounding box location), from 1 chair object with the location arranged in a 3x3 location matrix at 1.5-meter intervals with the assumption that it will get a more robust interpolation pattern. Each location with 4-way orientation.

Making the input image requires setting the location of a chair object in space above the floor surface with a 40x40 cm tile pattern (requirements that the pattern is sufficiently visible and easy to calculate up to a distance of 8 meters). The utilization of tile patterns is intended to facilitate setting the actual size of the location of the category object from the camera location. Objects in the image are on the floor where the accuracy of the object's location will be measured from the camera's location, arranged in a 3x3 pattern, and each location is 80 cm to the side and 160 cm in the direction of the camera. The dimensions of the effective surface area of the floor used are 240 x 600 cm square including the camera position to the nearest object location 200 cm away.

Each location is made of 4 orientations with an interval of 90 degrees. The pattern settings are set with the aim of being seen in the cellphone camera. This arrangement becomes a reference for the formation of a perspective grid and the creation of a primary dataset used for object detection testing and for determining the location of objects on the floor surface. Making the primary dataset produces 72 images annotated with the actual data bounding box.

In this section, we first discuss methods for estimating 3D locations from images in the form of object detection in the form of pre-trained models, namely: centerNet, SSD, and Yolov3 with Resnet50 and darkNet architectures from 2 MS-COCO and Pascal-VOC datasets. The following discusses the perspective grid concept for transforming from 2D points to 3D points on a surface plane.

The relationship between the camera direction and the vanishing point on the horizon line

(skyline), where the vanishing point is on a vertical line in the center of the image field (camera screen), then the vanishing point outside the image field will be on the vertical line (center of the earth towards the sky) and vertical straight plane (flat). There are 3 directions of view to see objects from above the surface normally in human view. (Figure 4).

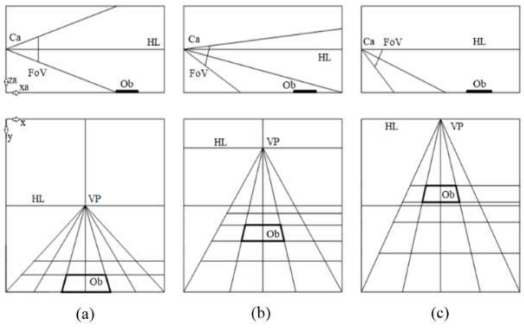


Figure 4. Alternate direction of the camera a. the missing point is on the horizon line of the screen, b. the missing point between the screen horizon line and the upper boundary line of the screen, c. the missing point is on the upper screen boundary line.

The approach to converting image point values into points on the floor plane of the room as shown in Figure 4 shows the pixel points in the image projected to be points on the floor plane in space with a value of z = 0. The missing point projection shows the x-ordinate on the bottom boundary line of the image and the projection of the floor line horizontally shows the y-ordinate on the side boundary line of the image.

Formation of a perspective grid pattern by taking a photo of the floor surface in a room with a 40x40 cm floor grid pattern. The camera settings are made at a height of 1.5 meters using a tripod. The camera uses a tripod as support. The camera is directed to the bottom half of the screen as far as 1.5 meters from the closest object.

3.2. Object Detection and Bounding Box with a pre-trained model

The use of the concept of transfer learning in pre-trained object detection models aims to predict the location of objects in the image (x', y'). The location of the object is searched using the lower left and lower right bounding box positions, assuming both are located on the floor surface (z'=0). The three object detection methods, namely CenterNet, SSD, and Yolov3 respectively use the MS-Coco and Pascal VOC datasets which contain 'chair' category objects with the object code of MS COCO being '56' and the object code of Pascal VOC being '8'.

Three pre-trained models to choose the best one. The detection method used as a model is practical and easy to use. Models sourced from the MXNet GluonCV toolkit library (https://cv.gluon.ai/) namely: SSD, CenterNet, and Yolov3. Implementation of pre-trained object detection models to determine the position of objects on the lower left and lower right of the bounding box from the object detection method on the floor surface. using three object detection pre-trained models from the MXNet GluonCV toolkit (https://cv.gluon.ai/), namely: SSD, CenterNet, and Yolov3.

Bounding boxes can be generated from object detection models, in this case using 3 models [SSD, CenterNet, Yolov3] using the Resnet and Darknet architectures from the 2 available datasets [MS COCO, Pascal VOC] for chair category objects. The Architecture Model and Dataset from the MXNet GluonCV toolkit. The method and dataset produce 6 kinds of combinations. The results of the 6 combinations to get the best-estimated value from testing the location of the chair category object using 36 test images, namely:

1. 'ssd_512_resnet50_v1_coco',
2. 'center_net_resnet50_v1b_coco'
3. 'yolo_darknet53_coco'
4. 'ssd_512_resnet50_v1_voc'
5. 'center_net_resnet50_v1b_voc'
6. 'yolo_darknet53_voc'

Average Precision (AP) is defined as the average detection precision at different draws and is usually evaluated categorically. AP is the most frequently used evaluation for object detection in recent years. Average AP (mAP) averaged across object categories is typically used as a metric to compare final performance across object categories

As a result of the popularity of the MS-COCO data set after 2014, there has been a shift in metrics as researchers began to pay more attention to the accuracy of bounding box locations. In addition to a fixed IoU threshold, at some IoU thresholds, the MS-COCO AP averages between 0.5 (coarse localization) and 0.95 (perfect localization), resulting in more accurate and possibly critical localization of objects.

Intersection over union (IoU) is used to measure the accuracy of object localization, checking whether the IoU between the prediction box and the basic truth box is greater than a predetermined threshold. In this study using IoU >= 0.5. Despite changes in the use of metrics, evaluation of object detection with VOC/COCO-based mAPs is still the most frequently used metric.

However, for studies that test an object, the AP category is an option.

3.3. Formation of the Perspective Grid

The process of transforming object coordinates in the image into 3D coordinates on the floor surface uses the perspective grid concept algorithm[31]. The perspective grid is formed by the intersection of two lines. These two lines are respectively represented by the equations a1x + b1y + c1= 0 and a2x + b2y + c2 = 0. The intersection points of the two lines are as follows:

$$a_1x + b_1y + c_1 = 0 \quad (1)$$

$$a_2x + b_2y + c_2 = 0 \quad (2)$$

$$x = \frac{(b_1c_2 - b_2c_1)}{(a_1b_2 - a_2b_1)} \quad (3)$$

$$y = \frac{(a_2c_1 - a_1c_2)}{(a_1b_2 - a_2b_1)} \quad (4)$$

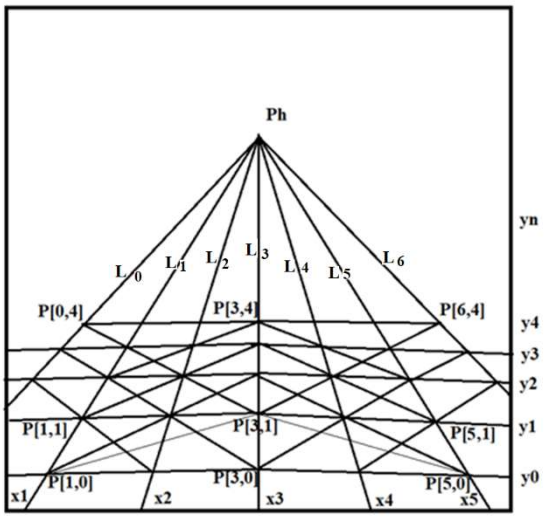


Figure 5. Perspective Grid Formation

The perspective grid approach is depicted on the floor surface built from one vanishing point through 6 reference points which are camera calibrations without geometry correction due to radial (barrel) distortion. Based on the floor grid from the camera view as reference points 1, 2, 3, 4, 5, and 6 namely: P[1,0], P[3,0], P[5,0], P[1,1], P[3,1], P[5,1] are defined manually directly on the image as shown in Figure 5. Then 3 guidelines are made from reference points 1 and 2, points 3 and 4, and points 5 and 6. The combination of intersections of the three lines will produce 3 intersection points which on average produce a vanishing point (ph).

Reference points 1, 2, and 3 are fixed, while points 4 (P[1,1]), 5 (P[3,1]), and 6 (P[5,1]) need to be redefined as follows: Find the point P[3,1] from the intersection of the line L1 from Ph to P[1,0] and the reference point 4 to reference point 5 produces P[1,1]. Then line L3: Ph to P[3,0] and line reference point 4 to reference point 5 gives P[3,1]'. Then line reference point 5 to reference point 6 with line L5: Ph to P[3,0] and line Ph to P[5,0] gives P[3, 1]" and point P[5,1]. And from P[3,1]' and P[3.1]", the midpoint is P[3,1].

Next line L2: Ph to the point of intersection at P[2,0.5] from the result of the intersection of line P[1,0] to point P[3,1] with line P[1,1] to point P[3, 0]. From the L2 line, it will produce points P[2,0] and P[2,1] respectively, which are obtained through the intersection with the line P[1,0] to P[3,0] and line P[1,1] to P[3,1].

To get the line L0: Ph to x0, it is necessary to find the point in P[0,2] which is formed from the intersection of the lines P[2,0] to P[1,1] and the lines P[3,2] to P[1,2]. Point P[3,2] is the result of the intersection of line L3 with line P[1,0] to P[2,1], and point P[1,2] is generated from the intersection of line L1 with line P[3,0] to P[2,1].

Likewise, taking the lines L4 and L6 can be done in the opposite direction (mirror from L3) in the same way as numbers 2 and 3, this results in points: P[4,0], P[4,1]. After that, an algorithm for forming a perspective grid is created so that all the points needed are obtained.

3.4. Object Locations in 2D Image

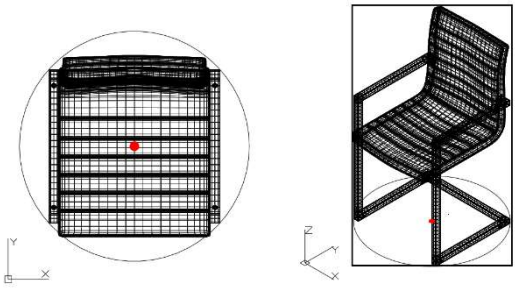


Figure 6. 3D CAD model with a center point (red circle) from the top view (left) and 3D view (right)

The location of an object in a 2D image is the center point of the object in the image as a determinant of the transformation to 3D. The shape of a real object in the representation of the CAD model is seen to be in the plane of the floor surface (z=0). Or from above the object with the object center point xo, yo set to be at the center point of the circle (red dot in figure 6). An illustration of a 3D chair being limited by a bounding box frame.

The transformation from 2D points on the image to 3D points on the floor surface, where the 2D points as the original points of the object are

obtained through the prediction line approach y based on a constant. The constant is obtained from the ratio of the proportion of the distance h from the bottom boundary line of the bounding box to the horizontal line. This horizontal line is formed through the midpoint of the object at point c (see figure 7) with the vertical distance from the vanishing point (x_v, y_v) to the bottom boundary endpoint of the bounding box (xL, yL). Prediction X (xc), obtained through the vertical line between the left and right bounding box boundaries on the prediction line y (yc).

$$x_c = \frac{(x_{LR} - x_{LL})}{2} \quad (5)$$

$$h = \frac{1}{f_n}(y_L - y_v) \quad (6)$$

$$y_c = y_L - h \quad (7)$$

The projection from the vanishing point through point c to the ground line (GL) will produce a predicted x value. The projection from point c to the vertical line will produce a predicted y value through the interpolation approach.

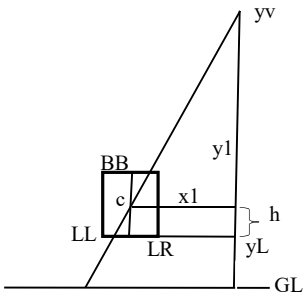


Figure 7. 2D object center point (c) and vanishing point perspective projection

By knowing the reference approach to the depiction of perspective projections where the boundary of the FoV (field of view) is the actual size scale. So that the image points on the floor plane can be projected from the vanishing point to the lower boundary line of the FoV plane.

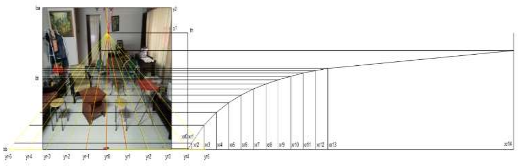


Figure. 8. Approach of two congruent triangles for the value of X_p and the interpolation equation for the value of Y_p.

The projection from the vanishing point to the lower boundary line of the image plane forms a triangle, therefore the pixel distance in the image can be equated with the distance on the floor plane through a similar triangle equation approach or a linear equation:

$$\frac{x_p}{x_{ip}} = \frac{x_{aw}}{x_{ia}} \quad (8)$$

$$x_p = \frac{x_{ip}}{x_{ia}} x_{aw} \quad (9)$$

Where is: x_p *is x Prediction*,
x_{aw} *is an Actual floor width is* 40*cm*
$x_{ip} =$ *x predictive image*
$x_{ia} =$ *actual floor width on the image*

Y prediction is obtained from calculating the predicted y value in the image and then projecting it to the y sum with the Newton interpolation approach. Newton's divided differences interpolation: [32]

$$f_n(x) = b_0 + b_1(x - x_0) + \cdots + b_n(x - x_0)(x - x_1) \ldots (x - x_{n-1}) \quad (10)$$

$$f_n(x) = b_0 + \sum_{i=1}^{n} b_i \prod x - x_{i-1} \quad (11)$$

Where:

$$b_0 = f(x_0) \quad (12)$$

$$b_1 = f[x_1, x_0] = \frac{f(x_1) - f(x_0)}{x_1 - x_0} \quad (13)$$

$$b_2 = f[x_2, x_1, x_0] = \frac{f[x_2, x_1] - f[x_1, x_0]}{x_2 - x_0} \quad (14)$$

$$\ldots$$

$$b_n = f[x_n, x_{n-1,\ldots}, x_1, x_0] = \frac{f[x_n, x_{n-1,\ldots}, x_2, x_1] - f[x_{n-1}, x_{n-2,\ldots}, x_1, x_0]}{x_n - x_0} \quad (15)$$

3.5. 3D Location Estimation Above Floor Level

Like 2D location estimation, based on the analysis of the perspective grid model from the vanishing point in the direction of the line to the lower boundary line of the image and the horizontal lines formed from the perspective grid to the vertical line of the image boundary.

The predicted coordinates consist of an x or x_p ordinate and a y or y_p ordinate. The x_p ordinate is obtained from the ratio of similar triangles [33] resulting from the projection of the vanishing point to the reference point. The yp ordinate is calculated by using the interpolation method. The interpolation approach uses Newton's divided methods.

4. EXPERIMENT RESULTS AND DISCUSSIONS

4.1. Object Detection and Bounding Box

The results of object detection testing using the Intersection of Union (IoU) between the actual bounding box and the predicted bounding box are shown in Figure 8, and the level of accuracy by

Journal of Theoretical and Applied Information Technology
30th June 2023. Vol.101. No 12

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

measuring the Average Precision (AP) from Lower Left and Lower Right Corner Accuracy.

The object detection approach uses 3 pre-trained models with 2 datasets containing chair objects each. All models are compared based on IoU and Precision values (1 object category) and the average distance of the lower left and lower right coordinates between the actual and predicted bounding boxes.

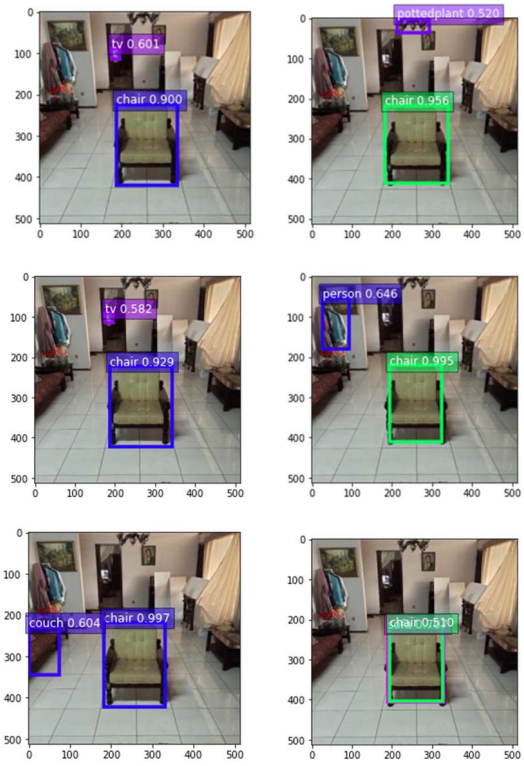


Figure. 9. Object Detection Result; (a) SSD, (b) CenterNet, (c) Yolov3. Kolom kiri MS-Coco dan kolom kanan Pascal VOC.

In table 1. The results of experiments on 36 test images of the n value indicate the number of correctly detected and positive (TP). the highest average IoU value uses the Yolov3 method with a COCO dataset of n as many as 32 images, namely 91.9% followed by the SSD method with a COCO of 89.1% with 32 images of TP value and the CenterNet method with a COCO of 87.07% with 31 images with TP value.

Table 1. Average IoU, AP, deviation from L&R Bottom of Bounding Box

Model	TP	Avg. IoU	AP	Avg_ dev L	Avg_ dev R
SSD-COCO	32	89.17%	93.94%	22.81	22.54
CenterNet-COCO	31	87.07%	91.12%	17.41	20.85
Yolov3-COCO	32	91.90%	96.50%	24.52	21.66
SSD-VOC	32	79.68%	84.27%	35.05	36.19
CenterNet-VOC	29	86.15%	96.45%	27.06	23.90
Yolov3-VOC	22	82.78%	91.06%	35.67	31.50

The Average Precision (AP) value of the number of images detected with the highest true positive was using the Yolov3 method with a COCO dataset of 96.50% and slightly different from the CenterNet method with a Pascal-VOC dataset of 96.45%, followed by the SSD method with a COCO dataset.

The results of the lowest average deviation of the lower left bounding box position and the smallest average deviation of the lower right bounding box position were produced by the CenterNet pre-trained model method with the COCO dataset, namely 17.41 cm and 20.86 cm. These results are from 31 images with the True-positive category from 36 test images.

So that the next implementation of object detection uses the Yolov3 method with the MS-COCO dataset to produce a predicted bounding box position. The 36 test images produced: 32 images in the True-positive category, 2 True-Negative images, and 2 False-Positive images, so the implementation uses 34 images with 2 clean data (TN).

In figure 10, a deviation value of 0 indicates the bounding box's vertical boundary coincides with the actual bounding box's vertical boundary. A deviation value of more than 0 indicates that the bounding box's vertical boundary is to the right of the actual vertical boundary, and vice versa.

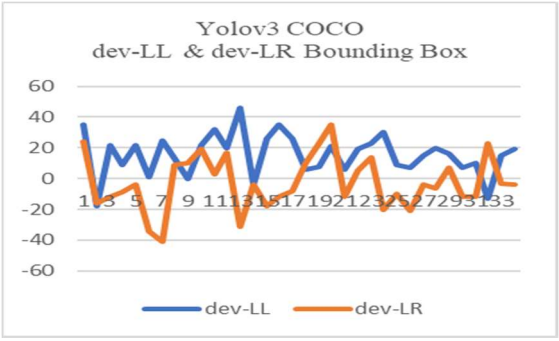
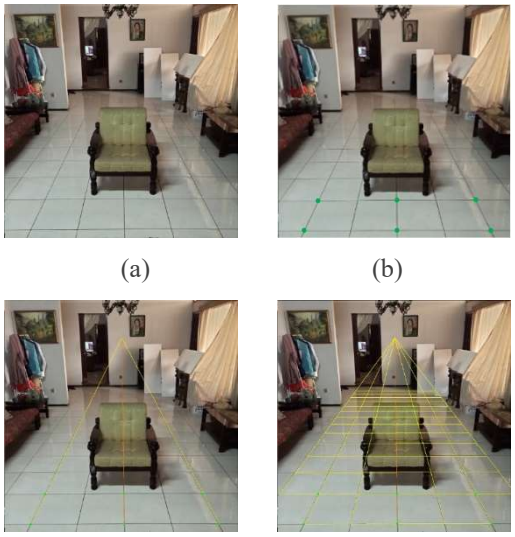


Figure. 10. Deviation Lower Left & Deviation Lower Right of Bounding Box using Yolov3 – COCO dataset

The graph shows the numbers showing the tendency of the prediction bounding box deviation to the right, especially the lower right point. This could be caused by the limitations of the Yolov3 model with MSCOCO, lighting effects, camera quality, and the location of objects in the room.

4. 2. Perspective Grid

The perspective grid implementation initially displays a photo scene of a chair object in space (figure 11. a), followed by manually defining 6 reference points (green color) (figure 11. b).



(a) (b)

(c) (d)

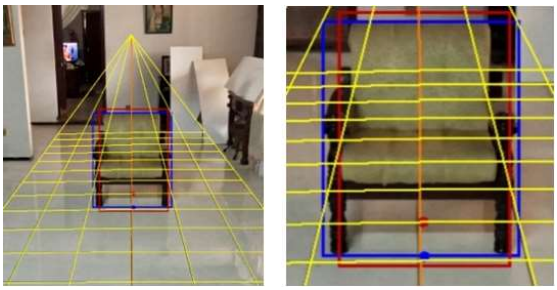
Figure. 11. Perspective Grid Result for testing

The six reference points must be exactly where the floor lines intersect. After that, three lines are formed, each of which is generated from two reference points. So that the three lines will produce three vanishing points, then be averaged to one vanishing point (figure 11. c). After that, correct the projection line from the vanishing point to the reference point, in the form of two yellow lines and one orange line. Furthermore, the process of forming a perspective grid is shown in Figure 11. d.

4.3. Predict Object Location with Perspective Grid

Determination of constant parameters with the approach of the 2nd order Newtonian interpolation equation from the vanishing point using the divisor constants for f0, f1, and f2 namely 8, 12, and 14 in the direction of x0, x1, and x2 in the form of distance values of 240, 400, 560 cm perpendicular to the direction camera.

The results of the left and right boundaries of the bottom bounding box are used as a reference for calculating the location of the center point of the chair object in the image. The location is located at the intersection of floor lines. Based on equations (5), (6), and (7) implemented using 34 images, the average error is obtained as shown in table 2.



(a) *(b)*

Figure. 12. (a) Perspective grid and Bounding Box Result with Yolov3 and MS-Coco, (b) 2D position (red point)

Table 2. Min, Average, MAE, Max value from Average dx, dy, dR

	dx	**dy**	**dR**
	cm	cm	cm
Min	0.02	1.01	1.37
Average	**0.47**	**1.08**	**6.47**
MAE	2.25	5.72	6.47
Max	6.29	17.93	18.03

The measurement error between the predicted location of the chair object compared to the actual location on the floor surface (table 2), namely: the average difference of 6.47-cm from the value of dx is 0.47-cm and dy is 1.08-cm. When compared to the dimensions of a chair object that is more than 50 cm (length, width, height), that is, dx is less than 0.94% and dy is less than 2.16%.

These results are consistent with the research objective of predicting the location of chair objects on the floor surface by achieving significant accuracy using a single image through the object detection approach and the perspective grid concept.

Table 3. dx, dy, & dR Average Error per Location

Matrix Location	**xa, ya**	**Adx**	**Ady**	**AdR**
	cm	cm	cm	cm
1,1	-80, 240	4.17	2.66	4.94
1,2	0, 240	2.01	4.71	5.12
1,3	80, 240	2.69	5.04	5.71
2,1	-80, 400	1.87	2.81	3.37

2,2	0, 400	2.22	5.90	6.30
2,3	80, 400	2.86	4.59	5.41
3,1	-80, 560	1.55	8.66	8.80
3,2	0, 560	1.46	7.05	7.20
3,3	80, 560	0.87	11.93	11.96

Viewed per location point in table 3, the biggest average difference for the x-axis is found at the location that is in the front row of the camera closest to the left, which is 4.17 cm. the biggest difference for the y-axis is in the location which is in the third row from the right camera. For the smallest average, the difference in distance between the predicted location and the actual location is 3.37 cm in the middle location on the left.

Table 4. dx, dy, & dR Average Error per Orientation

Orientation	dx	dy	dR
	cm	cm	cm
0^0	1.60	5.69	5.90
90^0	2.21	6.30	6.68
180^0	2.53	7.78	8.19
270^0	2.61	3.33	4.23

The largest average difference in distance between the predicted location and the actual location of 11.96 cm is in the third row on the right but has the smallest difference in x value, namely 0.87 cm. This shows that the change between locations is not linear.

This condition can be caused by the quality of the point density on the surface of the camera, differences in uneven lighting reception from the left in front of the object and the right behind the object which is more dominant and affects objects and shadows on the floor, and the determination of constant parameters is not quite right.

Viewed per object orientation (table 4), object orientation is facing the camera (0^0) with the smallest difference in x value and the second smallest difference in average distance. Object orientation towards the right from the camera (270^0) has the largest average difference but has the smallest average distance difference of 4.23 cm. Although the orientation is towards the left (90^0) as a reflection of the orientation to the right, the smallest average distance difference is in third place. This could be due to the difference in the direction of lighting from the right side of the camera which is more dominant.

5. CONCLUSIONS AND FUTURE WORK

In this paper, we have reviewed the estimation of object locations from single images using deep learning methods in recent years. Location of 3D objects to be used to construct 3D through a drawing of CAD models. This review focuses on different 2D to 3D transformation methods. We use the perspective grid concept approach by utilizing previously trained object detection methods to obtain predictions of object locations from the parameter bounding box.

In this work, we show how to determine the 2D location of a known object category from a single view, through a perspective grid approach to predict 3D locations, we transform 2D coordinates into 3D coordinates. Our method estimates a fairly accurate 3D location through 6 reference points in the image without additional depth values or the use of more than one image or stereo camera.

One future direction is to explore our method with more than one object category, automatically determining the constant parameters of the interpolation equation both for determining the predicted y values at locations in the image and for calculations on the floor surface. Another way is to explore our method in the video, which requires the use of temporal information and can enable the prediction of object location and velocity effectively in the future. Improve the level of accuracy by developing a category-specific object dataset as needed with sufficient numbers for training and testing the method of direct object detection (not pre-trained).

And the research requires accurate object location predictions including predictions of 3D (indoor) spatial shapes to be implemented in the field of robotics, autonomous cars, surveillance cameras, and mapping surveys using drones. Continuing research on the other side of pose estimation predicts the orientation of category objects in space from the camera direction based on a perspective grid.

REFERENCES

[1] H. Yi, *The application of auto CAD drawing technique in interior design*, vol. 842, no. 7. Springer International Publishing, 2019.

[2] Q. Xiaowen, "Information Sharing and Application of 3D Design Software in Interior Design," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 631, no. 5, 2019, doi: 10.1088/1757-899X/631/5/052011.

[3] S. Jin, Y. Zhang, T. Yamazaki, and Z. Jiang, "Automatic 3D CAD Model and 2D Drawings Generation in Construction Engineering," *J. Phys. Conf. Ser.*, vol. 1827, no. 1, 2021, doi: 10.1088/1742-6596/1827/1/012115.

[4] K. Chen, Y. K. Lai, and S. M. Hu, "3D indoor scene modeling from RGB-D data: A survey," *Comput. Vis. Media*, vol. 1, no. 4, pp. 267–278, 2015, doi: 10.1007/s41095-015-0029-x.

[5] S. Hoque, M. Y. Arafat, S. Xu, A. Maiti, and Y. Wei, "A Comprehensive Review on 3D Object Detection and 6D Pose Estimation with Deep Learning," *IEEE Access*, vol. 9, pp. 143746–143770, 2021, doi: 10.1109/ACCESS.2021.3114399.

[6] T. H. M. Siddique, Y. Rehman, T. Rafiq, M. Z. Nisar, M. S. Ibrahim, and M. Usman, "3D Object Localization Using 2D Estimates for Computer Vision Applications," *Proc. 2021 Mohammad Ali Jinnah Univ. Int. Conf. Comput. MAJICC 2021*, 2021, doi: 10.1109/MAJICC53071.2021.9526270.

[7] H. Izadinia, Q. Shan, and S. M. Seitz, "Im2Cad," *Proc. - 30th IEEE Conf. Comput. Vis. Pattern Recognition, CVPR 2017*, vol. 2017-Janua, pp. 2422–2431, 2017, doi: 10.1109/CVPR.2017.260.

[8] A. X. Chang, M. Savva, and C. D. Manning, "Learning spatial knowledge for text to 3D scene generation," *EMNLP 2014 - 2014 Conf. Empir. Methods Nat. Lang. Process. Proc. Conf.*, pp. 2028–2038, 2014, doi: 10.3115/v1/d14-1217.

[9] X. F. Han, H. Laga, and M. Bennamoun, "Image-based 3d object reconstruction: State-of-the-art and trends in the deep learning era," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 5, pp. 1578–1604, 2019, doi: 10.1109/TPAMI.2019.2954885.

[10] Z. Zou, Z. Shi, Y. Guo, and J. Ye, "Object Detection in 20 Years: A Survey," pp. 1–39, 2019, [Online]. Available: http://arxiv.org/abs/1905.05055.

[11] Y. Hold-Geoffroy *et al.*, "A Perceptual Measure for Deep Single Image Camera Calibration," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 2354–2363, 2018, doi: 10.1109/CVPR.2018.00250.

[12] J. H. Lee, "Camera calibration from a single image based on coupled line cameras and rectangle constraint," *Proc. - Int. Conf. Pattern Recognit.*, no. Icpr, pp. 758–762, 2012.

[13] R. A. Boby and S. K. Saha, "Single image based camera calibration and pose estimation of the end-effector of a robot," *Proc. - IEEE Int. Conf. Robot. Autom.*, vol. 2016-June, pp. 2435–2440, 2016, doi: 10.1109/ICRA.2016.7487395.

[14] M. Lopez, R. Mari, P. Gargallo, Y. Kuang, J. Gonzalez-Jimenez, and G. Haro, "Deep single image camera calibration with radial distortion," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2019-June, pp. 11809–11817, 2019, doi: 10.1109/CVPR.2019.01209.

[15] K. S. Choi, E. Y. Lam, and K. K. Y. Wong, "Automatic source camera identification using the intrinsic lens radial distortion," *Opt. Express*, vol. 14, no. 24, p. 11551, 2006, doi: 10.1364/oe.14.011551.

[16] H. S. Park, T. Shiratori, I. Matthews, and Y. Sheikh, "3D Trajectory Reconstruction under Perspective Projection," *Int. J. Comput. Vis.*, vol. 115, no. 2, pp. 115–135, 2015, doi: 10.1007/s11263-015-0804-2.

[17] G. Nakano, "Efficient DLT-based method for solving PnP, PnPf, and PnPfr problems," *IEICE Trans. Inf. Syst.*, vol. E104D, no. 9, pp. 1467–1477, 2021, doi: 10.1587/transinf.2020EDP7208.

[18] P. Wang, G. Xu, Y. Cheng, and Q. Yu, "A simple, robust and fast method for the perspective-n-point Problem," *Pattern Recognit. Lett.*, vol. 108, pp. 31–37, 2018, doi: 10.1016/j.patrec.2018.02.028.

[19] D. Campbell, L. Liu, and S. Gould, "Solving the Blind Perspective-n-Point Problem End-to-End with Robust Differentiable Geometric Optimization," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 12347 LNCS, pp. 244–261, 2020, doi: 10.1007/978-3-030-58536-5_15.

[20] G. Wang, F. Manhardt, F. Tombari, and X. Ji, "GDR-Net: Geometry-Guided Direct Regression Network for Monocular 6D Object Pose Estimation," 2021, doi: 10.1109/cvpr46437.2021.01634.

[21] Z. He, W. Feng, X. Zhao, and Y. Lv, "6D pose estimation of objects: Recent technologies and challenges," *Appl. Sci.*, vol. 11, no. 1, pp. 1–18, 2020, doi: 10.3390/app11010228.

[22] M. M. Fleck, "Perspective projection: the wrong imaging model," *Res. Rep.*, no. Kingslake 1992, pp. 95–01, 1995, [Online]. Available: http://www.cs.illinois.edu/~mfleck/my-papers/stereographic-TR.pdf.

[23] L. Shapiro, *computer vision*. 2001.

[24] M. T. Tran *et al.*, "Traffic flow analysis with multiple adaptive vehicle detectors and velocity estimation with landmark-based scanlines," *IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Work.*, vol. 2018-June, pp. 100–107, 2018, doi: 10.1109/CVPRW.2018.00021.

[25] Z. Tang, G. Wang, H. Xiao, A. Zheng, and J. N. Hwang, "Single-camera and inter-camera vehicle tracking and 3d speed estimation based on fusion of visual and semantic features," *IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Work.*, vol. 2018-June, pp. 108–115, 2018, doi: 10.1109/CVPRW.2018.00022.

[26] D. Bell, W. Xiao, and P. James, "Accurate Vehicle Speed Estimation from Monocular Camera Footage," *ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci.*, vol. 5, no. 2, pp. 419–426, 2020, doi: 10.5194/isprs-annals-V-2-2020-419-2020.

[27] N. Badariah, A. Mustafa, F. Bakri, and S. K. Ahmed, "Identification of Image Angle using Projective Transformation," pp. 408–413, 2014.

[28] S. Y. Edgerton, "Alberti's Perspective: A New Discovery and a New Evaluation," *Art Bull.*, vol. 48, no. 3–4, pp. 367–378, 1966, doi: 10.1080/00043079.1966.10790813.

[29] E. Baek and Y. Ho, "Depth Estimation and View Synthesis using Vanishing Point from Single View Image," pp. 82–85, 2014.

[30] Z. Zhou, F. Farhat, and J. Z. Wang, "Detecting Dominant Vanishing Points in Natural Scenes with Application to Composition-Sensitive Image Retrieval," *IEEE Trans. Multimed.*, vol. 19, no. 12, pp. 2651–2665, 2017, doi: 10.1109/TMM.2017.2703954.

[31] D. Rusjdi, E. Abdurachman, Y. Heryadi, and G. P. Kusuma, "Single Image Perspective Grid System on Floor Surface for 2D to 3D Transformation," in *2022 IEEE 8th International Conference on Computing, Engineering and Design (ICCED)*, 2022, pp. 1–6, doi: 10.1109/ICCED56140.2022.10010545.

[32] A. S. Shamloo and P. Hajagharezalou, "Interval Interpolation By Newtons Divided Differences," *J. Math. Comput. Sci.*, vol. 13, no. 03, pp. 231–237, 2014, doi: 10.22436/jmcs.013.03.05.

[33] P. Dvurechensky, A. Gasnikov, S. Omelchenko, and A. Tiurin, "Adaptive Similar Triangles Method: a Stable Alternative to Sinkhorn's Algorithm for Regularized Optimal Transport," 2017, [Online]. Available: http://arxiv.org/abs/1706.07622.