Alignment control using visual servoing and mobilenet single-shot multi-box detection (SSD): a review

Jayson Rogelio [a,b,1,*], Elmer Dadios [b,2], Argel Bandala [b,3], Ryan Rhay Vicerra [b,4], Edwin Sybingco[b,5]

[a] Department of Science and Technology- MIRDC, Taguig City, 1632, Philippines
[b] De La Salle University, Manila, 1004, Philippines
[1] jayson_rogelio@dlsu.edu.ph*; [2] elmer.dadios@dlsu.edu.ph; [3] argel.bandala@dlsu.edu.ph, [4] ryan.vicerra@dlsu.edu.ph;
[5] edwin.sybingco@dlsu.edu.ph
* corresponding author

ARTICLE INFO

Article history
Received August 31, 2021
Revised September 24, 2021
Accepted March 31, 2022
Available online March 31, 2022

Keywords
Visual Servoing
Deep Neural Network
Mobilenet-SSD
Object Detection
Alignment Control

ABSTRACT

The concept is highly critical for robotic technologies that rely on visual feedback. In this context, robot systems tend to be unresponsive due to reliance on pre-programmed trajectory and path, meaning the occurrence of a change in the environment or the absence of an object. This review paper aims to provide comprehensive studies on the recent application of visual servoing and DNN. PBVS and Mobilenet-SSD were chosen algorithms for alignment control of the film handler mechanism of the portable x-ray system. It also discussed the theoretical framework features extraction and description, visual servoing, and Mobilenet-SSD. Likewise, the latest applications of visual servoing and DNN was summarized, including the comparison of Mobilenet-SSD with other sophisticated models. As a result of a previous study presented, visual servoing and MobileNet-SSD provide reliable tools and models for manipulating robotics systems, including where occlusion is present. Furthermore, effective alignment control relies significantly on visual servoing and deep neural reliability, shaped by different parameters such as the type of visual servoing, feature extraction and description, and DNNs used to construct a robust state estimator. Therefore, visual servoing and MobileNet-SSD are parameterized concepts that require enhanced optimization to achieve a specific purpose with distinct tools.

1. Introduction

The surrounding physical environment involves extensive information perceived by the human vision and processed by the brain to effortlessly recognize objects and localize individuals when moving within an unfamiliar physical space. However, achieving the same robotic technologies capabilities is relatively challenging due to combining different innovations, including probabilistic localization, object detection, and object recognition, in computer vision and machine learning [1], [2]. Object recognition identifies a certain class of objects in an image, while object detection identifies an object's class and location in an image. However, numerous robotic grasping and manipulation approaches assume objects are known because recognition and detection are not real-time [3], [4]. Thus, unknown objects require enhanced analysis of the three-dimension structure and physical properties to infer a proper grasp. In this regard, realistic applications need to enhance the capability to deal with systematic and repeatable errors from inaccurate kinematic models and random errors from sensor noise or limited repeatability of the motors [3]. Reliable robotic systems require high optimization to work beyond open-loop execution in real-time with enhanced accuracy.

A comprehensive understanding of images requires a precise estimate of locations and concepts besides classification. This object detection process entails finding an object of interest and knowing its prior position in an image [2], [5]. The task involves skeleton, pedestrian, and face detection, which provide valuable information in autonomous driving, face recognition, human behavior analysis, and image classification [6], [7]. These Deep Neural Networks (DNNs) have been the powerful machine learning model constituting object detection, which precisely localizes objects rather than focusing only on image classification. However, object detection has inherent and significant limitations due to reliance on a pre-programmed trajectory and path, meaning the change in the environment or absence of an object makes the robot system unresponsive. Also, there is a relative lack of algorithms to ensure accurate object detection and alignment in the event of an occlusion. The integration of visual perception provides visual feedback, which constitutes a visual servoing responsible for vision-based control. The design of film handler alignment control for a portable x-ray system presents in this research and a review of merging visual servoing control with a deep neural network algorithm. The goal of this review paper was to illustrate and discuss a model design of film handler mechanism alignment control that combined vision and visual servoing. Likewise, presents the concept of visual servoing, feature extraction, and description, comparing position-based visual servoing and image-based visual servoing and the concept of the deep neural network specifically Mobilenet-SSD. This article aims to serve as a literature review of the current research and application of visual servoing and deep neural networks.

2. Method

In this section, alignment control will discuss using the design of a three (3) degrees of freedom (DOF) film handler alignment of a portable x-ray system as a sample model and a reference of discussion on the concept of visual servoing and deep neural network specifically MobileNet-Single Shot Detector (MobileNet-SSD). Likewise, it will discuss the step-by-step development of the film handler mechanism, focusing on the vision system and alignment controller. Also, the theoretical concept of feature extraction and description, visual servoing, and MobileNet-SSD. In a separate section, research studies related to the applications of visual servoing and DNN from year 2017- 2021 was summarized and presented MobileNet-SSD with other latest DNN networks. The development of X-ray film handler consists of several processes: (1) design using 3D CAD software, (2) procurement of parts and manufacture of film handler assembly, (3) integration of electronics hardware and actuators, and (4) evaluation for final mechanical adjustments [8]. The film handler of X-ray system with 3 DOF is shown in Fig. 1.

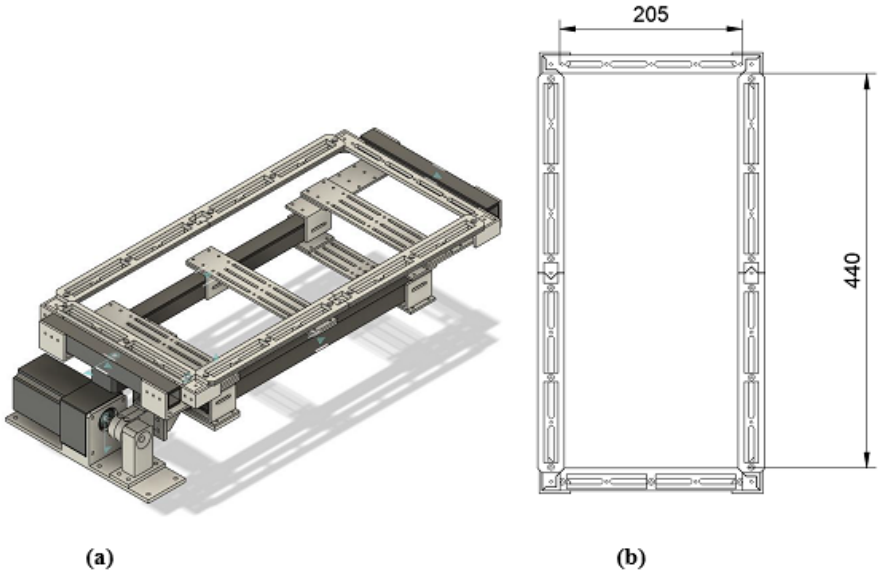

Fig. 1. (a) 3D CAD model of the X-ray film handler mechanism and (b) dimensions of the film handler frame [8]

Two perpendicular sliding mechanisms are used to achieve two-dimensional motion, which is operated by two stepper motors [8]. As indicated in Fig. 2 (a), one stepper motor will move the frame along the X-axis, while the other will move the frame along with the Y-axis Fig. 2 (b).

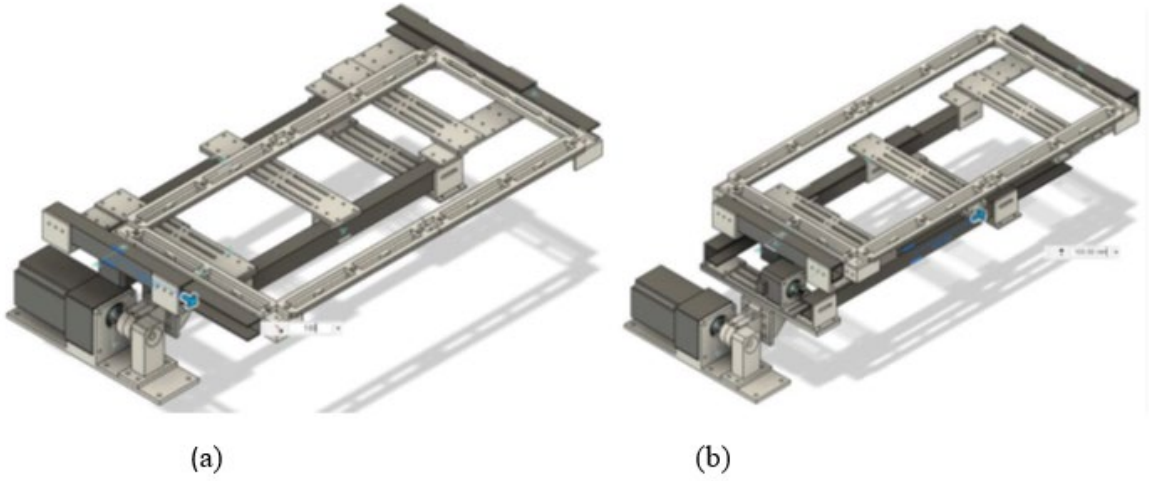

Fig. 2. Depiction of frame movement in two dimensions: (a) Frame displacement along X-axis and (b) frame displacement along Y-axis

A second stepper motor is mounted beneath the frame to adjust the total frame's pitch angle to the mobile platform. This allows the frame to be fastened between transports to prevent wind resistance from destroying the x-ray film. The alignment mechanism will first deploy the frame in a vertical position perpendicular to the direction of the X-ray source as the mobile platform approaches the suspicious object. Fig. 3 (a) shows the completely deployed film handler frame, and Fig. 3. (b) shows the partially deployed film handler frame.

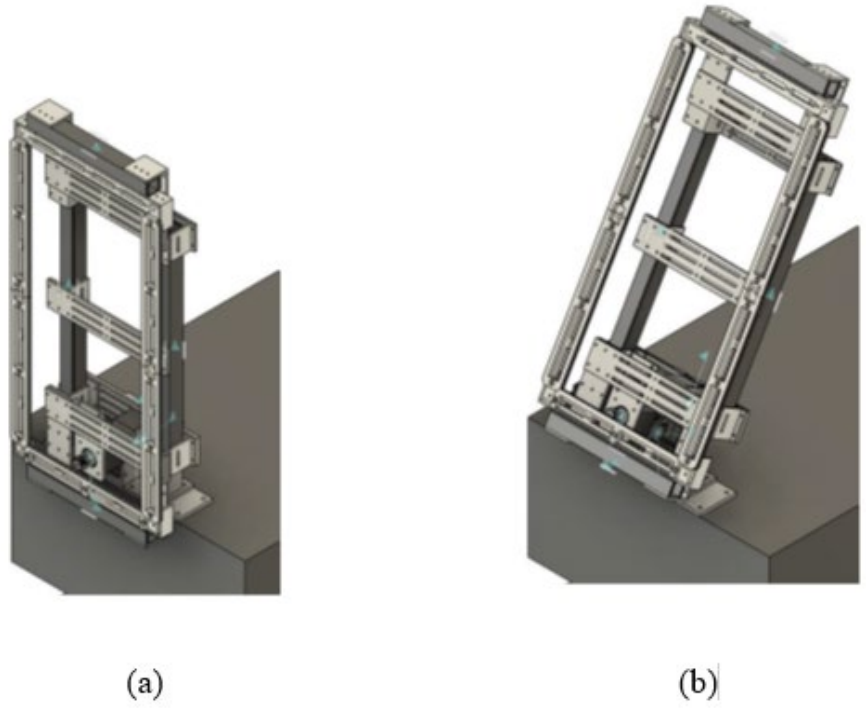

Fig. 3. Depiction of frame pitch angle: (a) fully deployed film handler frame and (b) partially deployed film handler frame

After the development of the x-ray film handler mechanism with adjustments being made to its mechanical structure, the development of the actuation control follows. Part of the alignment control development is properly attaching the motors, positioning sensors or limit switches, vision hardware, electrical wirings, and miscellaneous electronics hardware along with the frame structure. The challenge of this task is to properly place the components and electrical wiring so that the expected movement of the frame is not impeded. The drivers and controllers will be placed inside the mobile platform; thus, all wires will be re-routed underneath the frame to ensure that the electronics are well secured. Once the electronics hardware is in place, the microcontroller codes will be developed to interface with the stepper motor drivers, position encoders, and the main computer board. A simplified electronics schematic block diagram for the connection of these components is shown in Fig. 4.

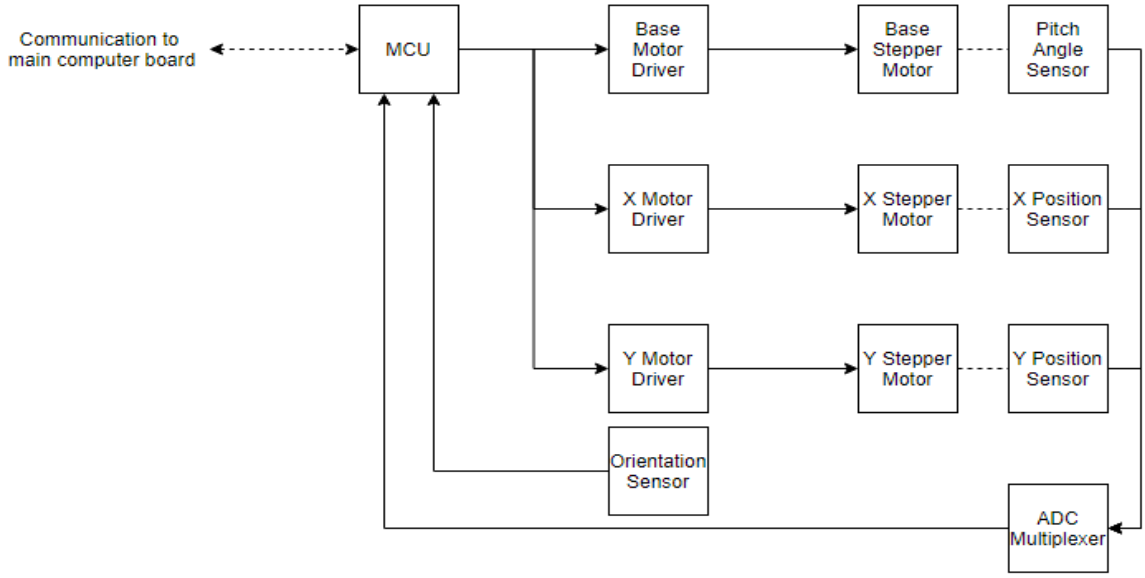

Fig. 4. Electronics schematic block diagram for the film handler alignment system

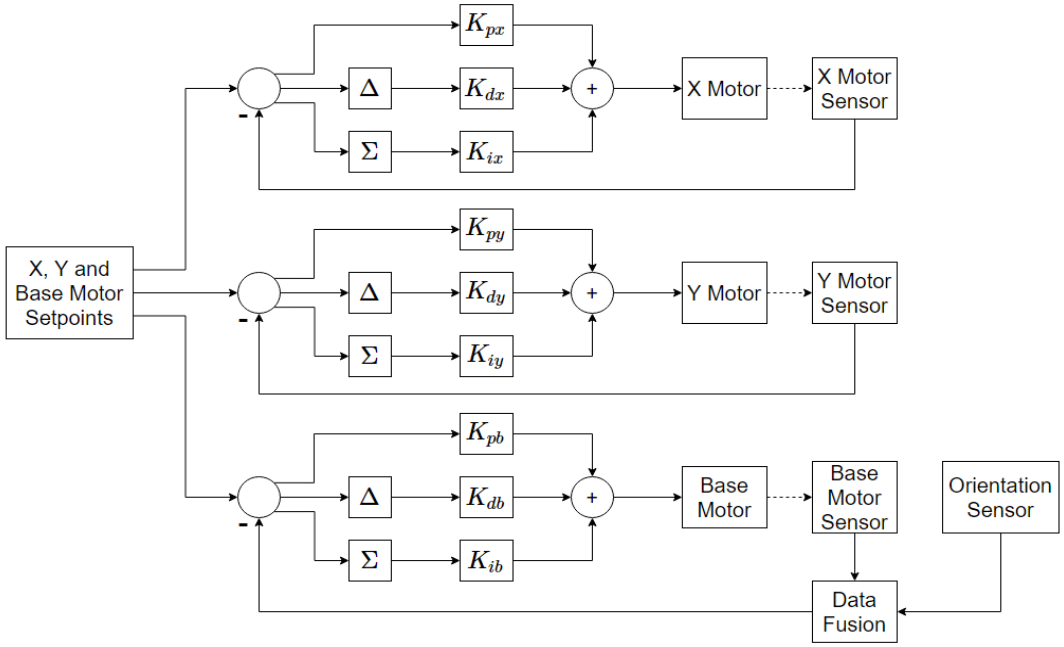

Fig. 5. Block diagram for x-ray film handler alignment system control

In designing the control for the three-stepper motors, a good understanding of feedback and control theory will be necessary. Fig. 5 shows the initial design of block diagram for the proposed control system.

The block diagram will also serve as a template for developing the required microcontroller code. Together with the sensors and actuators, the microcontroller system forms the inner control loop of the automated handler alignment system. Initially, the control system uses a proportional-integral-derivative (PID) controller to maintain the desired frame displacement. A series of independent motor tests will determine the PID parameters for each motor. With this system set up, its operation can stand alone, and thus, the output of the vision system needs to be a trajectory controller for this system. The setup helps simplify and isolate the problems, but it also eliminates the complexity associated with the interdependency of developing the system with machine vision. However, the possibility of interdependency will still be considered for this study when it would lead to further control optimization.

Fig. 6 shows the components of the vision system and the electronic interconnections between them. The Intel Realsense camera communicates directly with Jetson TX2. The advantage of the Intel Realsense camera is to offset the computation load from the main computer as it calculates the depth data internally and is sent to the main computer for further processing. This allows for faster development of target object reconstruction and focuses on developing the model needed for target detection and tracking. The illustration also shows how the camera is involved in a feedback loop since the camera is mounted on top of the frame mechanism, which is actuated by the stepper motors. The stepper motors are indirectly moved by the outcome of the vision system process.

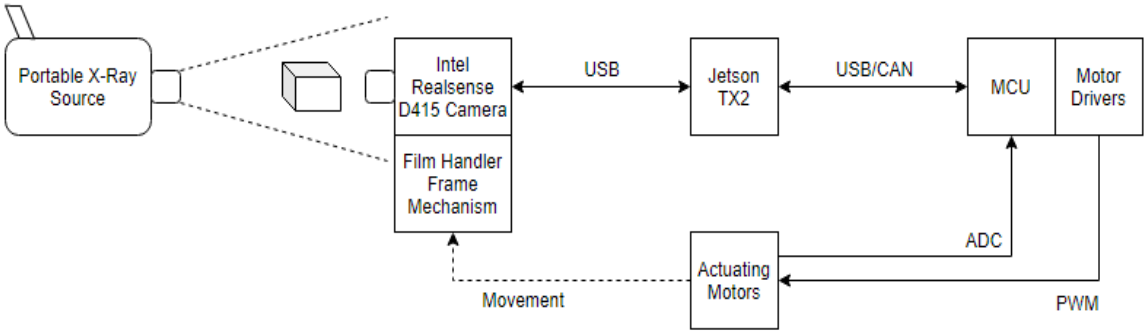

Fig. 6. Overview of the components of a vision system and their interconnections

Moreover, Fig. 7 shows the process flowchart of the vision system. The RGB+D visual data from the D415 camera will contain 4 separate layers per frame: red, green, blue, and depth components. The MobileNet neural network architecture will serve as a base model for object detection because it provides speed up through depth-wise convolutions. Another reason for selecting the said architecture is that the system developed by this study has limited processing power. The object detection model will be trained to detect the portable x-ray source on several environmental conditions. The feature extraction model will be used to track important features from the object and will use the same network architecture as a base network. The output of the object detection model will be a binarized mask of the portable x-ray source, while the feature extraction model will generate global feature vectors. The camera's pose to the portable x-ray is computed by combining the depth data with the extracted mask and features. Together with the trajectory generation process, the pose estimation process block will form a 3D space reconstruction that will identify the position of the portable x-ray source in the 3D Cartesian space used for the alignment control.

The trajectory generation process will also track the portable x-ray source from frame to frame. The occlusion prediction process will take care of occlusions, and it will work hand-in-hand with the trajectory generation process to produce an output coordinate for the alignment control. Thus, the study will determine the algorithms for the pose estimation, occlusion prediction, and trajectory generation processes. The film handler mechanism alignment system development is a combination of a vision system and alignment controller, which develop independently. Once the alignment system is fine-tuned for optimal response and the vision system has reached satisfactory accuracy for object detection and tracking, the development of the required visual servoing algorithm proceeds.

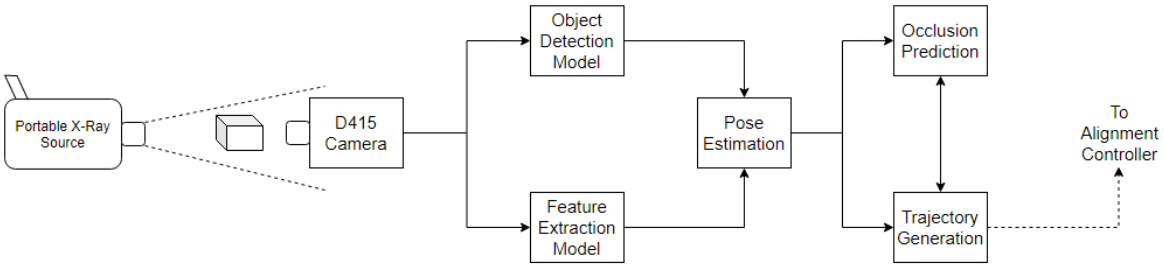

Fig. 7. Block diagram and process flowchart of the vision system

The sample in Fig. 8 shows the Position-Based Visual Servoing (PBVS) algorithm with the vision system as an external trajectory controller will be implemented. Evaluations will be taken, serving as a benchmark for the next model adjustments. The aim is to develop a model that will achieve better accuracy than the common externally-controlled-PBVS algorithm. This could mean that the PID controller could be modified to enhance the response further or that the vision system feedback will become part of the internal feedback required to drive the motors.

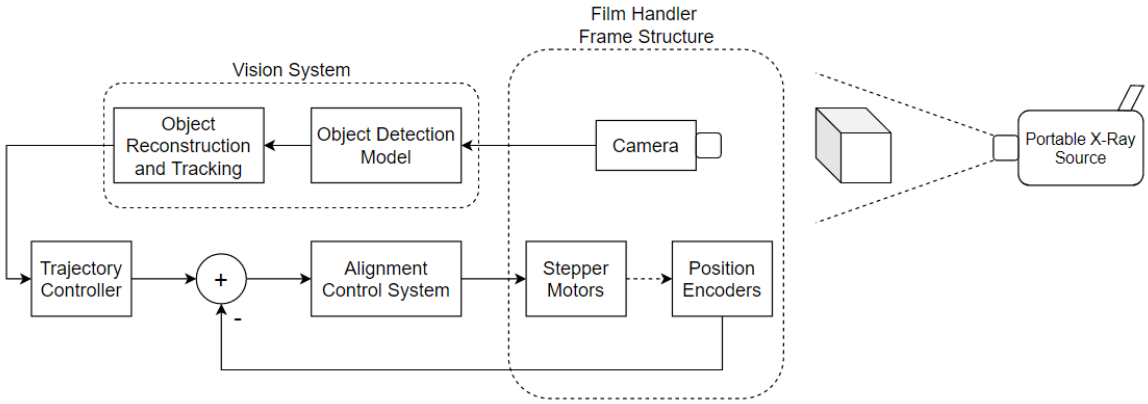

Fig. 8. Externally-controlled PBVS scheme for the film handler alignment system

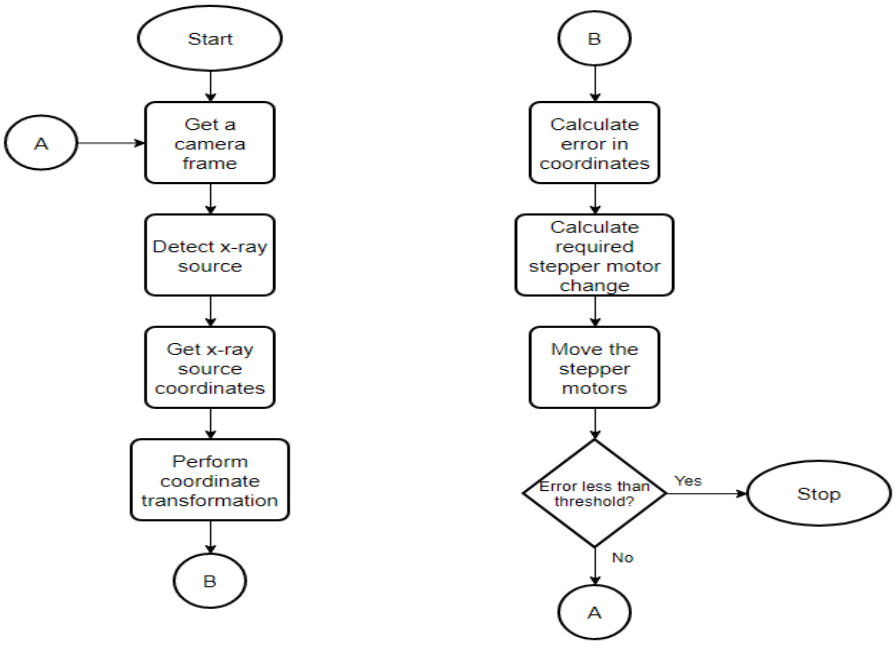

Fig. 9. Simplified process flowchart for the integrated alignment and vision control system

The camera has to get a frame or snapshot of its environment and attempt to detect the portable x-ray source and lock onto it as the mobile platform moves. The coordinates of the tracked features will be used to reconstruct the target's pose relative to the camera. Appropriate coordinate transformations must be performed to develop the metrics to be measured. Coordinate transformation involves transforming from the camera frame to the portable x-ray source frame or the center of an x-ray film frame. The errors will then be calculated and shall be used to calculate the required motion of the stepper motors to bring the x-ray film center in alignment with the portable x-ray aperture axis and ensure that the x-ray film is also perpendicular to the aperture axis. The process is iterated until the alignment error is less than the threshold value, as shown in Fig. 9.

3. Theoretical Framework

3.1. Feature Extraction and Description

A static and dynamic scene description is a crucial aspect of computer vision [9]. Thus, feature detection and description are essential in robotic system control to identify interest points for describing image content, such as blobs, ridges, corners, and edges [10]. The concept's primary goal is to describe the semantics of actions and behavior through object detection, analysis, and tracking [11]. In this regard, novel, robust, and automated features detection and description algorithms with high accuracy and performance are emerging to manage access control, perform statistical analysis, detect suspicious actions, track vehicles, and identify military targets [10], [12]. However, service robots are significantly unreliable in non-controlled and highly dynamic scenes due to enhanced demand for motion close to the object intended for manipulation, planning, and execution that drive the arm and refine the previous stage's final position relative to the correct position [13].

Consequently, feature extraction and detection need to include a servoing scheme, either image-based or pose-based, for handling uncertainties. The continued enhancement of robotic technology aims at attaining feature extraction and description with advanced human-robot communication. In this context, Human-Robot-Interaction (HRI) becomes highly intuitive with a high level of natural modality, whereby robotic systems understand and execute human orders, such as skeleton motions, facial expressions, eye-gaze, or hand gestures, in the absence of direct touch sensors [14]. However, enhanced HRI requires motion sensors to differentiate object movements and those caused by the sensors themselves to effectively articulate background modeling to create an appropriate background for each frame [15]. Moreover, trajectory classification is highly essential for computing feature points that differentiate trajectories belonging to the same objects from the background. In this context, small objects lack appearance information for differentiating them from the background or similar categories, creating numerous possibilities and an enhanced need for accurate localization [16]. Therefore, feature detection and description are relatively complex due to the need to consider extensive dynamism, despite improving algorithms and approaches.

3.2. Visual Servoing

Robotic manipulators are highly popular industrial processes due to the enhanced demand for autonomy. Autonomous object manipulation involves vision-based sensory systems for repetitive pick-and-place in dynamic environments, such as kitting, bin-picking, product packaging, path planning, and trash detection [17], [18]. Meanwhile, the integration of visual servoing in robotic systems improved flexibility by providing enhanced adaptation capabilities of neural networks and continuous feedback to the learning [19]. Thus, visual servoing is responsible for predicting image coordinates' trajectories and fostering the movement of sensors during manipulation. The concept eliminates the need for geometry information and allows robotic arms to reach for moving targets by following a trajectory without stopping at the specified intermediate points [19]. Hence, visual servoing is gaining popularity in unstructured environments to offer diverse services to people while interacting and exploring their environments rather than following a predefined path. Robotic systems with eye-to-hand or one or two eye-in-hand configurations utilize either position-based visual servoing (PBVS) or image-based visual servoing (IBVS). PBVS computation of three dimensions (3D) Cartesian errors, meaning that the model

requires perfect calibration and eye-in-hand modeling to generate a strong sensitivity to noise perturbations [20]. In contrast, IBVS involves feature points on an image plane for regulating the robotic pose, while underlying mapping is conducted to describe differences in end-effector velocities and visual features in a Cartesian space, as shown in Fig. 10. Although the model does not require 3D target restructuring, the accurate calculation of a mapping matrix is essential for successful implementation [21]. PBVS and IBVS utilize two feedback loops: an outer loop for controlling the error vector between the feature vectors and an internal loop for controlling the sensor's speed [21]. However, some manipulators have hybrid control systems that minimize the IBVS feature error in image space and PBVS' log depth ratio. Therefore, visual servoing provides definite models of handling uncertainties of the single-shot detector (SSD).

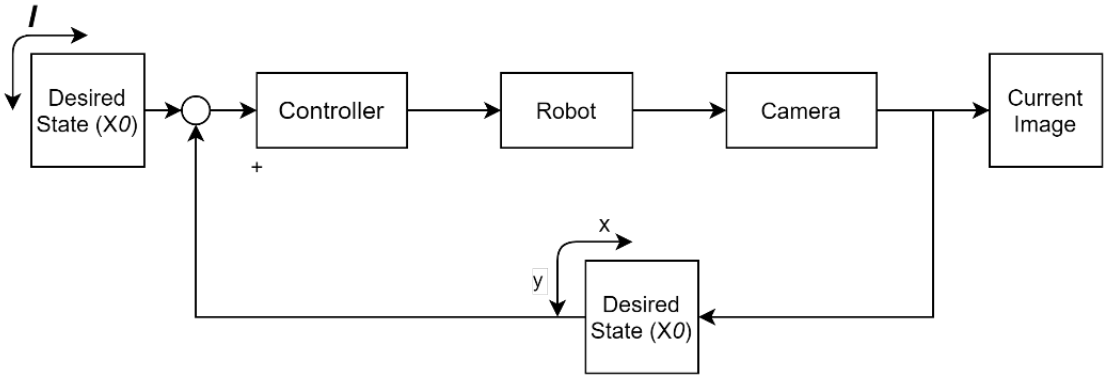

Fig. 10. Typical Block Diagram used in Image-Based Control [20]

In PBVS, $C\xi_T$ is estimated and is defined by the target's pose relative to the camera. Pose estimation needs a good estimate of the geometry of the target, intrinsic parameters of the camera, and the features of the observed image plane. The desired relative pose relative to the target is designated as ${}^{C^*}\xi_T$ and the motion required to move the camera from its initial $C\xi$ to ξ^*c pose is designated to be ξ_Δ with an actual pose of the unknown target ξ_T shown in Fig. 11. Therefore, the pose network can be written according to [22] is;

$$\xi_\Delta \oplus {}^{X^*}\xi_T = {}^{X^*}\widehat{\xi_T} \tag{1}$$

where ${}^{C}\widehat{\xi_T}$ is the estimated pose of the target relative to the camera and can be re-arranged as

$$\xi_\Delta = {}^{X}\xi_T \ominus {}^{X^*}\widehat{\xi_T} \tag{2}$$

which is the camera motion required to achieve the desired relative pose.

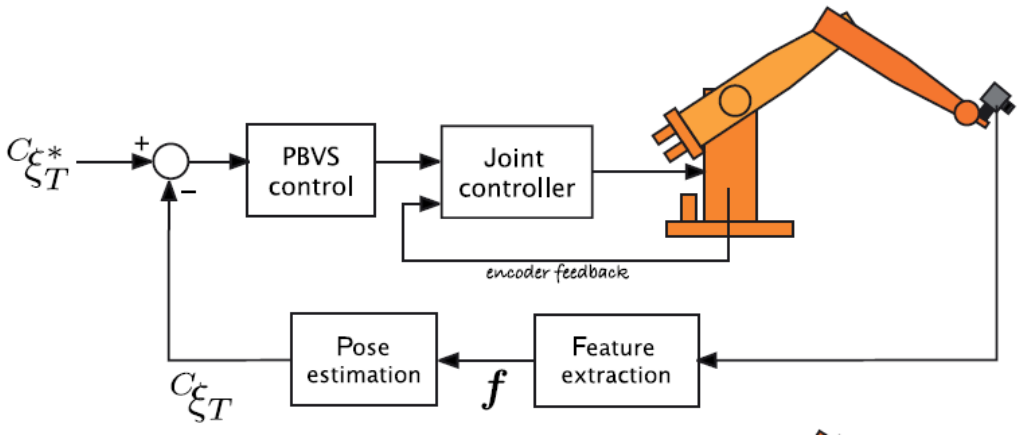

Fig. 11. Position-based visual servo [23]

Computer vision is highly effective in enabling PBVS to identify a manipulated object's pose relative to a robot's pose. However, the model is not effective in the differentiation of distinct components. In this context, effective alignment control requires iterative closest point (ICP) for constraining Euclidean distance between the closest source points and target clouds, planar segmentation for removing scene noise from point clouds, and principal component analysis for identifying the lower order linear subspaces located in the high dimensional datasets [22], [23]. The model also involves contour searching to describe an object's topological structure and cross-correlation for measuring the vector displacement of one object relative to another. Nonetheless, the high efficiency of PBVS requires enhanced design space exploration using genetic algorithms, randomized search, simulated annealing, and Bayesian optimization to identify DNN architecture that delivers high accuracy [24]. DNN improves the system's robustness against dynamic noises and constructs a robust state estimator with high precision [25]. Therefore, effective utilization of PBVS requires enhanced optimization to achieve the desired result and improved reliability. IBVS utilizes a combination of image processing methods to achieve accuracy in feature extraction. The model's primary goal is to utilize the eye-in-hand robotics system to move the robot end-effector to the desired pose from the current pose [26]. The approach utilizes numerous image processing methods, including the HOG-based method, SIFT-based method, contour-based method, and RGB-based method, which rely on characteristic aspects of an image to ensure the sensor image is equivalent to the target object [20]. IBVS allows observation of depth and its integration into visual servoing for enhanced accuracy. In this context, different methods describe shape information using height, width, rotation angle, arc length, and parameter equations but fail to consider slight shape changes, making visual servoing incomplete [27].

Consequently, IBVS integrates triangular surface mesh, piecewise model, active growing neural gas network, or adaptive contour feature to provide shape information during visual detection and tracking. The model is adaptive to translational dynamics and parameter uncertainty by effectively integrating thrust constant, desired feature depth, and mass [28]. Thus, IBVS provides enhanced visual servoing by considering an object's contours or depth, generating 3D feature extraction and description.

3.3. MobileNet

MobileNets are lightweight DNNs characterized by low-power and low-latency models that meet the demand of resource-constrained use cases. Debnath et al. consider the concept as an efficient CNN architecture used in mobile vision and embedded applications due to their lightweight [29]. MobileNets (Fig. 12) is founded on streamlined architecture, utilizing pointwise and depth-wise convolutions [29].

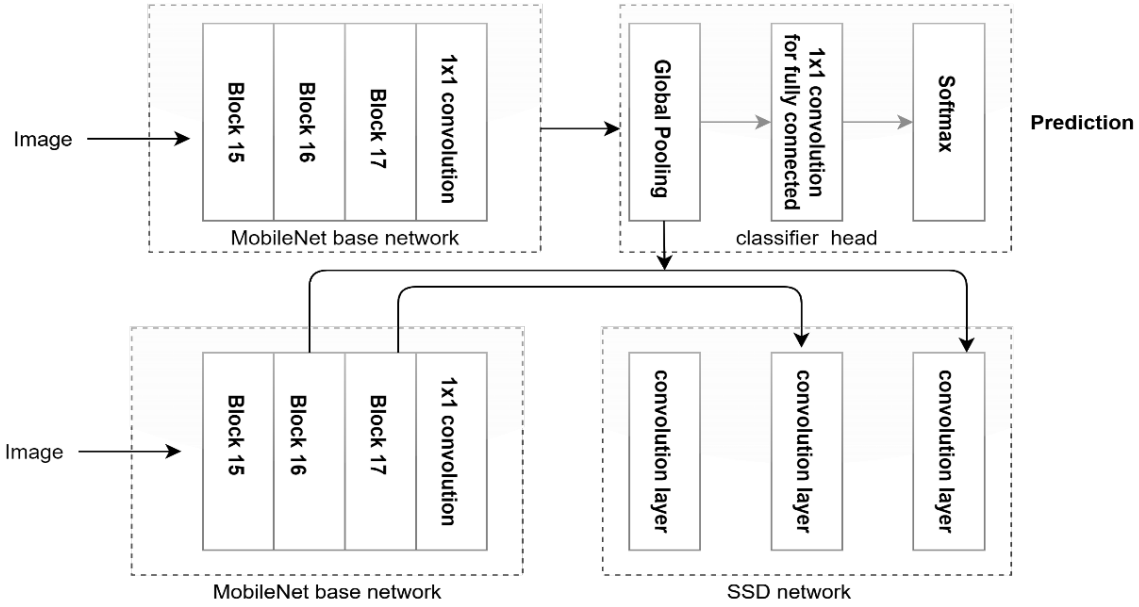

Fig. 12. Sample of MobileNet and MobileNet-SSD Architecture [29]

The main difference between streamlined architecture and MobileNets is the convolution operation, where MobileNets involve filtering and combining inputs and outputs as two steps rather than one step. Nonetheless, the network is relatively fast due to the reduced numbers of parameters, whereby 3x3 depth-wise separable convolutions reduce computations by nine times despite a fractional reduction inaccuracy. MobileNets reduce computation in the first few layers by embracing depthwise separable convolutions and inception models. The embedded pointwise convolution factorizes standard convolution into a 1×1 convolution and depth-wise convolution, which reduces computation and model size [30]. Therefore, MobileNets institute autonomous behavior into systems to reduce execution and cognitive burden on users by facilitating remote inspection and package delivery, besides effectively surveying hostile environments.

3.4. MobileNet-SSD

In recent literature, interest was growing in building small and efficient neural networks for mobile vision applications using modern deep learning models to perform visual servoing such as object detection. The SSD approach is based on a convolutional feed-forward network that produces a fixed-size collection of bounding boxes and scores for the presence of object class instances in those boxes, followed by a non-maximum detection step to produce the final detections [31], [32]. In a recent study of W. Liu et al, the SSD algorithm is faster than YOLO and significantly more accurate than, in fact, slower techniques that perform explicit region proposals and pooling, such as Faster R-CNN [33]. Fig. 8 shows the comparison between SSD and YOLO. The SSD model adds several feature layers to the end of a base network, which predicts the offsets to default boxes of different scales and aspect ratios and their associated confidences. SSD with a 300 × 300 input size significantly outperforms its 448 × 448 YOLO counterpart inaccuracy on the VOC2007 test while improving the speed, as shown in Fig. 13.

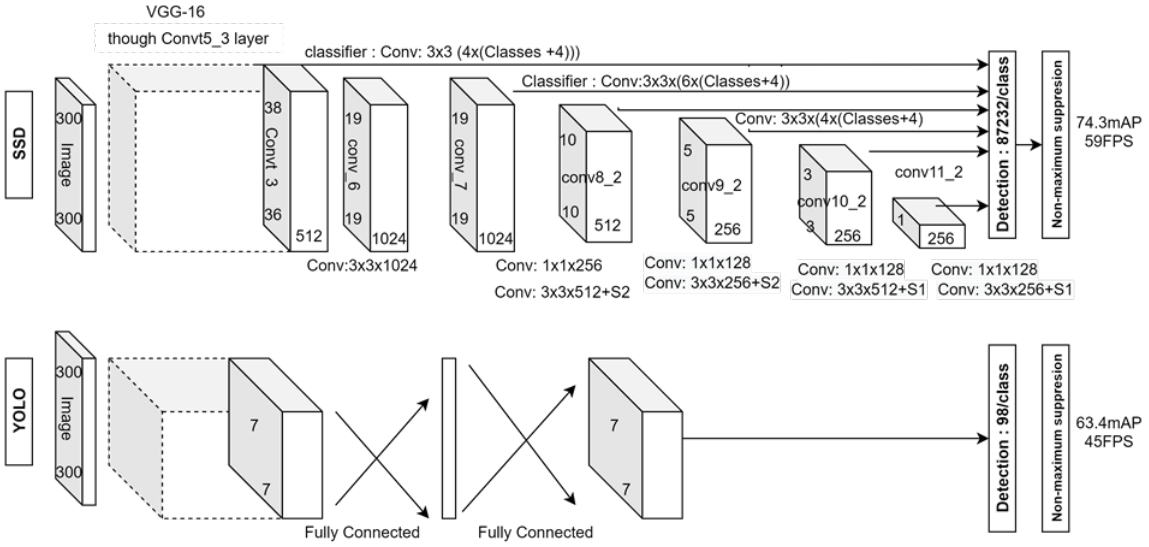

Fig. 13. A comparison between two single-shot detection models: SSD and YOLO [41]

MobileNet CNN model was created specifically for embedded vision applications, such as autonomous driving, face categorization, and object identification [34]. In comparison to ordinary convolutions, MobileNET used depthwise separable convolutions to lower the calculation cost to around one-eight [35]–[38]. Recently there have been several developments in MobileNET, such as MobileNET V2 [35], Enhanced Hybrid MobileNET [36], and ShuffleNet [38]. MobileNet V2 is an enhanced architecture as compared to MobileNet V1 in terms of model size. The architecture uses linear bottleneck blocks [36], [39] in the standard convolutional layers. The usage of successive layers is significant in preventing too much data from being destroyed. It largely reduces the model size but decreases the accuracy compared to baseline MobileNet [35]. The Enhanced Hybrid MobileNet [36] is a new architecture proposed to improve further the performance of the MobileNet V1 [40] model. A new hyperparameter called depth multiplier [36] was introduced, and the average pool layer was replaced by the Max pooling layer with stride two or Fractional max-pooling [36], [39] with stride 1.4. Various

combinations of the width multiplier [35], [39], [40] and the depth multiplier [35] were tried out with Max pool (stride=2) or Fractional max pool (stride=1.4). Some models have higher accuracy, while some have a smaller size compared to the baseline MobileNet. Lastly, the ShuffleNet [38] is a unique architecture based on MobileNet V1, which utilizes 1 × 1 pointwise group convolutions and the channel shuffle method.

Furthermore, A Younis et al. developed a solution that combines MobileNET and the SSD framework, as shown in Fig. 14. For Single Shot Multi-Box Detector (SSD), the MobileNET was employed as a deep learning pre-trained model. The method developed demonstrated good object detection accuracy at a processing speed of 14 frames per second, making it suitable for all cameras that can only process at 6 frames per second [41]. Similarly, X. Hu et al. employed MobileNET-SSD MicroScope to increase license plate detection accuracy, increase anti-interference capability, and deploy real-time on the RK3399 mobile device [42]. S. Zhao et al. developed MobileNET-SSD for real-time data capture of target recognition of a suggested signal-switching model based on deep learning for dynamic regulation of pedestrian traffic. The proposed algorithm model consists of the Long and short term memory model (LSTM), the object detection model, the MobileNET-SSD, and the decision model. Likewise, Dembys et al. [43] developed an approach for recognizing and estimating 3D poses of objects using deep learning that runs on embedded hardware. The MobileNET SSD is used to recognize and track objects of interest in the scene.

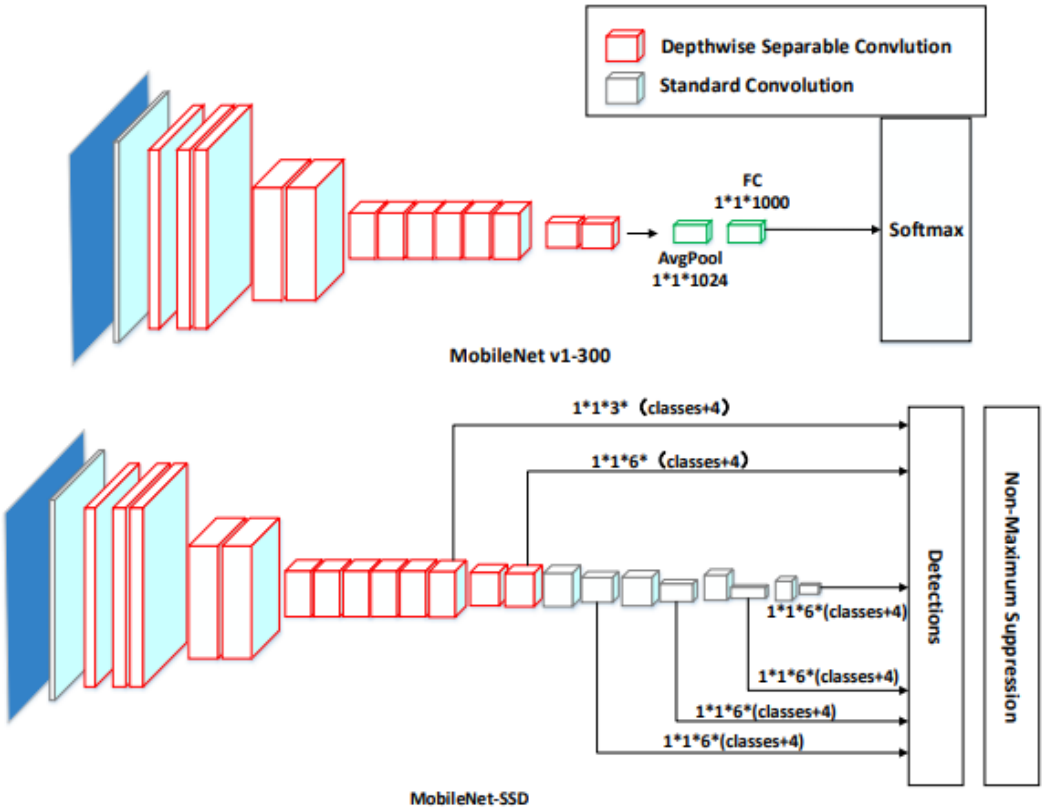

Fig. 14. MobileNET-SSD Model [41]

In contrast, in the pose estimation phase, the algorithm uses stereo correspondences to 3D reconstruct the spatial coordinates of multiple Oriented FAST and Rotated BRIEF (ORB) features in the recognized object's bounding box. M. Razavi et al. [44] used MobileNet-SSD for a real-time object detection system in an enclosed environment on the Jetson TK1, which improves performance by changing the networks' convoys and dividing tasks between the central and the graphic processor. Also, Rahul and Nair successfully utilized MobileNET SSD and stereo vision camera system to detect and identify objects and estimate their distances from the camera.

4. Results and Discussion

Research on robot vision servo technology has emerged significantly in real-time position and precise monitoring of static and moving targets. Different robot cameras are mounted to identify target features, process feature information, and provide visual input to manipulate the robot. A visual servo system is a complex, non-linear system that includes vision and controls. It also includes work on image processing (detection and recognition), robot kinematics and dynamics, and various control algorithms [45]. In 1979, Hill and Park implemented visual servoing based on the literature to differentiate it from the previous robot control algorithm. It typically adopts a static visual servo algorithm to collect the images and process them accordingly, and then the target position is determined by measurement [45]. Image servoing is used in various applications such as mobile navigation systems, auto-tracking, and object manipulation [46]. These general applications are object manipulation that involves either cooperative or non-cooperative object detection, image processing, segmentation, recognition, serving, alignment, or grasping for known objects. The usual visual servo function is to position and align the robot manipulator with the target. The usual task of this device is to track or maintain a constant distance or position between the robot and the moving target. For both instances, image information is used as a processing input to transfer the robot to its desired position. As above, it changes the robot's location in real-time and performs accurate tracking or positioning to complete the task. Other applications of visual servoing and deep neural networks are listed in Table 1.

Table 1. Previous Work and Application of Visual Servoing and DNN

Application	Method	Advantage	Disadvantage	Ref.
Autonomous robot with 7 DOF robotic grasping in unstructured and dynamic environment	Real-time application in autonomous robotic grasping using CNN for training dataset, Cornell Grasping Dataset (CGD) and Kinova Gen3 for robotic manipulator	CNN is simple with small parameters but with more visual information using data augmentation	Insufficiency in depth information for grasping	[47]
Visual servoing with 6 DOF control of robotic manipulator with low data information	Experiment on 6DOF Yaskawa Motoman MH5, visual servoing approach and CNN with Data Augmentation	Simple and feasible to use VS, CNN with data augmentation	Difficulty in mapping for from 2D image to the 3D space	[34]
IBVS control approach with robust state estimation for robot manipulation	Jacobian identification with Kalman filtering techniques	-The method compensates the state-estimation errors of Kalman filtering with NN. - The method does not require intrinsic and extrinsic parameters of the camera. -The hand-eye does not require calibration during robot manipulation.	Robot control of motion and position relies on visual feedback. - The method avoids calibration errors rather than solving them. - The proposed system is unstable due to dynamic noise with change in a large region	[20]
Adaptive visual servoing for describing shape information	Bezier-curve-feature-based method and NURBS-curve feature-based method	- The methods fit the contour of the object. - Effective for providing feedback on the shape information of the object to the eye-in-hand visual servoing system. - Allow representation of desired shape or angle.	Cubic Bezier curve is not effective for representing complex curves due to the generation of a huge computational burden. - Bezier curves are not robust for regular and symmetric curves. - Bezier curves lack the ability of local shape modification.	[51]

Table 1. (Cont.)

Application	Method	Advantage	Disadvantage	Ref.
Fuzzy neural network controller for a six-degrees-of-freedom robot manipulator	Simulation and Takagi–Sugeno fuzzy inference	-The method provides time efficiency, accuracy, and fast stability. - Enhances accuracy of the image preprocesses. - The method does not require computation of the inverse interaction matrix	Interference adversely affects the accuracy of the proposed system. - The method has a significant range of bias for the coordinates. - Features that are not within the camera's field of view increase instability.	[27]
Controlling a wheeled mobile robot equipped with a robotic manipulator	Simulation of feed-forward neural network	- Identifies a reliable image compression. - Reduces errors between original and reconstructed images used to control a wheeled mobile robot - Provides compressed images that can be used directly for segmentation purposes in visual servoing	The feed-forward neural network model does not entirely eliminate errors. - The use of a single layer of NN reduces effectiveness and reliability compared to vector quantization NN, Hebbian learning rule, and back-propagation.	[52]
Kinematic control of a manipulator with an eye-in-hand camera	Kinematic control of a manipulator with an eye-in-hand camera	-The control of joint angle and velocity enhances the safety of the manipulator during the visual servoing process. - Provides real-time manipulability optimization of redundant manipulators. - The model remedies the position error accumulation in traditional recurrent NN approaches	The simulation only involved PUMA 560 robot manipulator. - The model did not evaluate moving objects or uncertainty in the image Jacobian. The model may become relatively expensive with the addition of robot manipulators to handle complex tasks and applications.	[53]
ImageNet classification	Experiments on resource and accuracy tradeoff	MobileNets show strong performance compared to other popular models on ImageNet classification. MobileNets are effective for a wide range of applications and use cases - Utilizes factorization that reduces computation and model size	The models are suitable for building lightweight deep NNs. - MobileNet models were only trained in TensorFlow. - The models are subject to overfitting.	[54]
Visual servo control approach for the leader-follower platooning system based on homography	Homography matrix and simulation	-The method provides visual tracking by estimating the projective transformation. -The use of the entries of the homography matrix to estimate the velocity of the leading robot reduces computational cost. -The method involved control variables which increased the robustness of the platooning systems	- The simulation and experimentation used a virtual robot generated according to the homography and leader robot. - The simulation is based on the assumption that mobile robots drive on flat roads. - The construction of virtual robots requires prior knowledge of desired distance.	[55]

Table 1. (Cont.)

Application	Method	Advantage	Disadvantage	Ref.
IBVS for docking autonomous underwater vehicle	IBVS and simulation using Simulink™	-Present visual information as a reliable sensing mechanism in an underwater environment. - Enhances single camera of pin-hole model by integrating image processing and vision guidance controller	Pose estimation fails when features are out of the camera view or are missing due to malfunction or occlusion. - The success of the method relies on the proper projection of the target on the image IBVS relies on the motion of features on image plane	[56]

Table 2 illustrates that visual servoing and a deep neural network were successfully implemented in a robotic manipulator with three degrees of freedom (DOF) to seven degrees of freedom (DOF). Visual servoing and DNN, for example, were effectively applied in a 7 DOF robotic manipulator real-time grasping by E. Godinho et al. The work used a modest number of DNN parameters with data augmentation and visual servoing to tackle the challenge of grasping moving objects and improve object detection accuracy. J. Liu et al. used visual servoing and DNN for a 6-DOF robotic manipulator to simplify picture feature extraction and non-linear estimate relationships between 2D space in traditional visual servoing. Table 2 shows on the other hand, indicates that Mobilenet-SSD outperformed other advanced models in the ImageNet classification challenge and when compared to a variety of factors. The MobileNet-SSD network requires less computation and has fewer parameters because the accuracy varies less. This feature makes it easy to deploy on mobile devices and allows it to perform target identification tasks locally without using networking functions such as cloud services, lowering the road network system's overall processing capacity.

Table 2. Selection of Eigenvector [57]

Framework Resolution	Model	MAP	Billion Mult-Adds	Million Parameters
SSD 300	deeplab-VGG	21.2%	34.9	33.1
	Inception-V2	22.0%	3.8	13.3
	MobileNet	19.3%	1.2	6.8
Faster-RCNN300	VGG	22.9%	64.3	138.5
	Inception V2	15.4%	118.2	13.3
	MobileNet	16.4%	25.2	6.1
Faster-RCNN600	VGG	25.7%	149.6	138.5
	Inception V2	21.9%	129.6	13.3
	MobileNet	19.8%	30.5	6.1

5. Conclusion

The increased monitoring and surveying of dynamic environments increase the demand for robotic systems that can work in unstructured environments. As a result, the systems need to include SSD because the environment is continually changing or the sensors (cameras) are in motion. The use of visual servoing and mobilenet provide reliable tools and models for manipulating robotic systems, including instances where occlusion is present. In this context, effective alignment control significantly rely on the reliability of visual servoing and DNN, shaped by different parameters, such as IBVS and PBVS, used for feature extraction and description and DNNs used to construct a robust state estimator. Moreover, visual servoing and mobilenet are parameterized concepts that require enhanced optimization to achieve a specific purpose with distinct tools. This review paper presents the design of the film handler mechanism by integrating vision system and deep neural network. Also, it highlights the advantages of using visual servoing and DNN in robotic manipulation. Compared with other sophisticated models of object detection with little variation in accuracy, DNN such as Mobilenet-SSD requires less computation and has fewer parameters, increasing its efficiency.

Acknowledgment

The authors would like to acknowledge the support of the Science Education Institute and Metals Industry Research and Development Center of the Department of Science and Technology (DOST) of the Philippines and the Electronics and Communications Engineering Graduate Studies of the De La Salle University, Philippines.

Declarations

Author contribution. All authors contributed equally to the main contributor to this paper. All authors read and approved the final paper.
Funding statement. None of the authors have received any funding or grants from any institution or funding body for the research.
Conflict of interest. The authors declare no conflict of interest.
Additional information. No additional information is available for this paper.

References

[1] P. Durdevic, "A Deep Neural Network Sensor for Visual Servoing in 3D Spaces," 2020, doi: 10.3390/s20051437.

[2] J. A. C. Jose *et al.*, "Categorizing License Plates Using Convolutional Neural Network with Residual Learning," in *2019 4th Asia-Pacific Conference on Intelligent Robot Systems (ACIRS)*, 2019, pp. 231–234, doi: 10.1109/ACIRS.2019.8935997.

[3] X. Gratal, J. Romero, J. Bohg, and D. Kragic, "Visual servoing on unknown objects," *Mechatronics*, vol. 22, no. 4, pp. 423–435, Jun. 2012, doi: 10.1016/j.mechatronics.2011.09.009.

[4] R. R. P. Vicerra *et al.*, "A multiple level MIMO fuzzy logic based intelligence for multiple agent cooperative robot system," in *TENCON 2015 - 2015 IEEE Region 10 Conference*, 2015, pp. 1–7, doi: 10.1109/TENCON.2015.7372985.

[5] Y. J. Lee and A. Yilmaz, "Real-time object detection, tracking, and 3D positioning in a multiple camera setup," *ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci.*, vol. II-3/W2, no. November, pp. 31–35, Oct. 2013, doi: 10.5194/isprsannals-II-3-W2-31-2013.

[6] Z. Zhao, P. Zheng, S. Xu, and X. Wu, "Object Detection With Deep Learning: A Review," *IEEE Trans. Neural Networks Learn. Syst.*, vol. 30, no. 11, pp. 3212–3232, Nov. 2019, doi: 10.1109/TNNLS.2018.2876865.

[7] C. L. C. Bual, R. D. Cunanan, R. A. R. Bedruz, A. A. Bandala, R. R. P. Vicerra, and E. P. Dadios, "Design of Controller and PWM-enabled DC Motor Simulation using Proteus 8 for Flipper Track Robot," in *2019 IEEE 11th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM)*, 2019, no. 1, pp. 1–5, doi: 10.1109/HNICEM48295.2019.9072736.

[8] J. P. Rogelio *et al.*, "Modal Analysis, Computational Fluid Dynamics and Harmonic Response Analysis of a 3D Printed X-ray Film Handler for Assistant Robotic System using Finite Element Method," in *2020 IEEE 12th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM)*, 2020, pp. 1–6, doi: 10.1109/HNICEM51456.2020.9400014.

[9] R. L. Galvez, A. A. Bandala, E. P. Dadios, R. R. P. Vicerra, and J. M. Z. Maningo, "Object Detection Using Convolutional Neural Networks," in *TENCON 2018 - 2018 IEEE Region 10 Conference*, 2018, pp. 2023–2027, doi: 10.1109/TENCON.2018.8650517.

[10] E. Salahat and M. Qasaimeh, "Recent advances in features extraction and description algorithms: A comprehensive survey," in *2017 IEEE International Conference on Industrial Technology (ICIT)*, 2017, pp. 1059–1063, doi: 10.1109/ICIT.2017.7915508.

[11] R. L. Galvez, E. P. Dadios, A. A. Bandala, and R. R. P. Vicerra, "YOLO-based Threat Object Detection in X-ray Images," in *2019 IEEE 11th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM)*, 2019, pp. 1–5, doi: 10.1109/HNICEM48295.2019.9073599.

[12] A. A. Bandala *et al.*, "Development of Leap Motion Capture Based - Hand Gesture Controlled Interactive Quadrotor Drone Game," in *2019 7th International Conference on Robot Intelligence Technology and Applications (RiTA)*, 2019, pp. 174–179, doi: 10.1109/RITAPP.2019.8932800.

[13] J. R. Sanchez-Lopez, A. Marin-Hernandez, E. R. Palacios-Hernandez, H. V. Rios-Figueroa, and L. F. Marin-Urias, "A Real-time 3D Pose Based Visual Servoing Implementation for an Autonomous Mobile Robot Manipulator," *Procedia Technol.*, vol. 7, pp. 416–423, 2013, doi: 10.1016/j.protcy.2013.04.052.

[14] Yali Li, Shengjin Wang, Qi Tian, and Xiaoqing Ding, "Learning Cascaded Shared-Boost Classifiers for Part-Based Object Detection," *IEEE Trans. Image Process.*, vol. 23, no. 4, pp. 1858–1871, Apr. 2014, doi: 10.1109/TIP.2014.2307432.

[15] M. Yazdi and T. Bouwmans, "New trends on moving object detection in video images captured by a moving camera: A survey," *Comput. Sci. Rev.*, vol. 28, pp. 157–177, May 2018, doi: 10.1016/j.cosrev.2018.03.001.

[16] K. Tong, Y. Wu, and F. Zhou, "Recent advances in small object detection based on deep learning: A review," *Image Vis. Comput.*, vol. 97, p. 103910, May 2020, doi: 10.1016/j.imavis.2020.103910.

[17] and A. S. S. S. A. O. M. F. Demirci, "Deep Learning-Based Object Classification and Position Estimation Pipeline for Potential Use in Robotized Pick-and-Place Operations," *Robotics*, vol. 9, no. 3, p. 63, Aug. 2020, doi: 10.3390/robotics9030063.

[18] T. W. Teng, P. Veerajagadheswar, B. Ramalingam, J. Yin, R. Elara Mohan, and B. F. Gómez, "Vision Based Wall Following Framework: A Case Study With HSR Robot for Cleaning Application," *Sensors*, vol. 20, no. 11, p. 3298, Jun. 2020, doi: 10.3390/s20113298.

[19] D. Kuhn, J. L. Buessler, and J. P. Urban, "Neural approach to visual servoing for robotic hand eye coordination," in *Proceedings of ICNN'95 - International Conference on Neural Networks*, 1995, vol. 5, pp. 2364–2369, doi: 10.1109/ICNN.1995.487731.

[20] Z. Xungao, X. Min, G. Jiansheng, Z. Xunyu, and P. Xiafu, "Robot manipulation using image-based visual servoing control with robust state estimation," in *2018 Chinese Control And Decision Conference (CCDC)*, 2018, pp. 445–449, doi: 10.1109/CCDC.2018.8407174.

[21] D. E. Touil, N. Terki, A. Aouina, and R. Ajgou, "Intelligent Image-based-Visual Servoing for Quadrotor Air Vehicle," in *2018 International Conference on Communications and Electrical Engineering (ICCEE)*, 2018, pp. 1–7, doi: 10.1109/CCEE.2018.8634553.

[22] R. Mahony, P. Corke, and F. Chaumette, "Choice of image features for depth-axis control in image based visual servo control," in *IEEE/RSJ International Conference on Intelligent Robots and System*, 2009, vol. 1, pp. 390–395, doi: 10.1109/IRDS.2002.1041420.

[23] W. Lin, A. Anwar, Z. Li, M. Tong, J. Qiu, and H. Gao, "Recognition and Pose Estimation of Auto Parts for an Autonomous Spray Painting Robot," *IEEE Trans. Ind. Informatics*, vol. 15, no. 3, pp. 1709–1719, Mar. 2019, doi: 10.1109/TII.2018.2882446.

[24] L. Shi, "An Object Detection and Pose Estimation Approach for Position Based Visual Servoing," *Electr. Control Commun. Eng.*, vol. 12, no. 1, pp. 34–39, Jul. 2017, doi: 10.1515/ecce-2017-0005.

[25] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, "SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5MB model size," pp. 1–13, 2017, doi: 10.48550/arXiv.1602.07360.

[26] H. Rezatofighi, N. Tsoi, J. Gwak, I. Reid, and S. Savarese, "Generalized Intersection over Union : A Metric and A Loss for Bounding Box Regression," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 658–666. doi: 10.1109/CVPR.2019.00075

[27] W. Pan, M. Lyu, K. Hwang, M. Ju, and H. Shi, "A Neuro-Fuzzy Visual Servoing Controller for an Articulated Manipulator," *IEEE Access*, vol. 6, pp. 3346–3357, 2018, doi: 10.1109/ACCESS.2017.2787738.

[28] F. Wang, F. Sun, J. Zhang, B. Lin, and X. Li, "Unscented Particle Filter for Online Total Image Jacobian Matrix Estimation in Robot Visual Servoing," *IEEE Access*, vol. 7, pp. 92020–92029, 2019, doi: 10.1109/ACCESS.2019.2927413.

[29] H. Xie, A. F. Lynch, K. H. Low, and S. Mao, "Adaptive Output-Feedback Image-Based Visual Servoing for Quadrotor Unmanned Aerial Vehicles," *IEEE Trans. Control Syst. Technol.*, vol. 28, no. 3, pp. 1034–1041, May 2020, doi: 10.1109/TCST.2019.2892034.

[30] B. Debnath, M. O'Brien, M. Yamaguchi, and A. Behera, "Adapting MobileNets for mobile based upper body pose estimation," in *2018 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, 2018, no. November 2019, pp. 1–6, doi: 10.1109/AVSS.2018.8639378.

[31] S. Khalid, T. Khalil, and S. Nasreen, "A survey of feature selection and feature extraction techniques in machine learning," in *2014 Science and Information Conference*, 2014, no. July, pp. 372–378, doi: 10.1109/SAI.2014.6918213.

[32] D. Sinha and M. El-Sharkawy, "Ultra-thin MobileNet," in *2020 10th Annual Computing and Communication Workshop and Conference (CCWC)*, 2020, pp. 0234–0240, doi: 10.1109/CCWC47524.2020.9031228.

[33] M. Hassaballah, A. A. Abdelmgeid, and H. A. Alshazly, *Image Feature Detectors and Descriptors*, vol. 630. Cham: Springer International Publishing, 2016. Available at: Google Scholar.

[34] J. Liu and Y. Li, "Visual Servoing with Deep Learning and Data Augmentation for Robotic Manipulation," *J. Adv. Comput. Intell. Intell. Informatics*, vol. 24, no. 7, pp. 953–962, Dec. 2020, doi: 10.20965/jaciii.2020.p0953.

[35] M. Sandler, M. Zhu, A. Zhmoginov, and C. V Mar, "MobileNetV2: Inverted Residuals and Linear Bottlenecks," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 4510–4520. doi: 10.1109/CVPR.2018.00474

[36] H. Chen and C.-Y. Su, "An Enhanced Hybrid MobileNet," in *2018 9th International Conference on Awareness Science and Technology (iCAST)*, 2018, no. September 2018, pp. 308–312, doi: 10.1109/ICAwST.2018.8517177.

[37] J. Guo, "Network Decoupling: From Regular to Depthwise Separable Convolutions," in *arXiv preprint arXiv:1808.05517*, 2018, pp. 1–12, doi: 10.48550/arXiv.1808.05517.

[38] J. S. Xiangyu Zhang, Xinyu Zhou, Mengxiao Lin, "ShuffleNet: An Extremely Efficient Convolutional Neural Network for Mobile Devices," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 6848–6856. Available at: Google Scholar.

[39] B. Graham, "Fractional max-pooling," in *arXiv preprint arXiv:1412.6071*, 2014, pp. 1–10. Available at: Google Scholar.

[40] A. G. Howard *et al.*, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," in *arXiv preprint arXiv:1704.04861*, 2017, doi: 10.48550/arXiv.1704.04861.

[41] A. Younis, L. Shixin, S. Jn, and Z. Hai, "Real-Time Object Detection Using Pre-Trained Deep Learning Models MobileNet-SSD," in *Proceedings of 2020 the 6th International Conference on Computing and Data Engineering*, 2020, pp. 44–48, doi: 10.1145/3379247.3379264.

[42] M. Papoutsidakis, K. Kalovrektis, C. Drosos, and G. Stamoulis, "Design of an Autonomous Robotic Vehicle for Area Mapping and Remote Monitoring," *Int. J. Comput. Appl.*, vol. 167, no. 12, pp. 36–41, Jun. 2017, doi: 10.5120/ijca2017914496.

[43] W. Zhang and G. Zhang, "Image Feature Matching Based on Semantic Fusion Description and Spatial Consistency," *Symmetry (Basel).*, vol. 10, no. 12, p. 725, Dec. 2018, doi: 10.3390/sym10120725.

[44] W. Rahmaniar and A. Hernawan, "Real-Time Human Detection Using Deep Learning on Embedded Platforms: A Review," *J. Robot. Control*, vol. 2, no. 6, pp. 462–468, 2021, doi: 10.18196/jrc.26123.

[45] D. Kragic and H. I. Christensen, "Survey on Visual Servoing for Manipulation," *Comput. Vis. Act. Percept. Lab. Fiskartorpsv*, vol. 15, pp. 1–58, 2002. Available at: Google Scholar.

[46] Y. Wang, D. Ewert, R. Vossen, and S. Jeschke, "A Visual Servoing System for Interactive Human-Robot Object Transfer," *J. Autom. Control Eng.*, vol. 3, no. 4, pp. 277–283, 2015, doi: 10.12720/joace.3.4.277-283.

[47] E. G. Ribeiro, R. de Queiroz Mendes, and V. Grassi, "Real-time deep learning approach to visual servo control and grasp detection for autonomous robotic manipulation," *Rob. Auton. Syst.*, vol. 139, p. 103757, May 2021, doi: 10.1016/j.robot.2021.103757.

[48] D. Cabecinhas, S. Bras, R. Cunha, C. Silvestre, and P. Oliveira, "Integrated Visual Servoing Solution to Quadrotor Stabilization and Attitude Estimation Using a Pan and Tilt Camera," *IEEE Trans. Control Syst. Technol.*, vol. 27, no. 1, pp. 14–29, Jan. 2019, doi: 10.1109/TCST.2017.2768515.

[49] J. Liu and Y. Li, "An Image Based Visual Servo Approach with Deep Learning for Robotic Manipulation," in *The 6th International Workshop on Advanced Computational Intelligence and Intelligent Informatics*, 2019, pp. 1–6, doi: 10.48550/arXiv.1909.07727.

[50] Q. Bateux *et al.*, "Training Deep Neural Networks for Visual Servoing To cite this version : HAL Id : hal-01716679 Training Deep Neural Networks for Visual Servoing," 2018. doi: 10.1109/ICRA.2018.8461068

[51] H. Wang, S. Member, B. Yang, J. Wang, and X. Liang, "Adaptive Visual Servoing of Contour Features," vol. 23, no. 2, pp. 811–822, 2018. doi: 10.1109/TMECH.2018.2794377

[52] V. Nicolau, M. Andrei, and G. Petrea, "Aspects of Image Compression using Neural Networks for Visual Servoing in Robot Control," pp. 2–6, 2017. doi: 10.1109/ISEEE.2017.8170627

[53] Y. Zhang, S. Li, B. Liao, L. Jin, and L. Zheng, "A Recurrent Neural Network Approach for Visual Servoing," no. July, pp. 614–619, 2017. doi: 10.1109/ICInfA.2017.8078981

[54] A. G. Howard, B. Chen, and W. Wang, "MobileNets : Efficient Convolutional Neural Networks for Mobile Vision MobileNets : Efficient Convolutional Neural Networks for Mobile Vision Applications," no. October, 2020. Available at: Google Scholar.

[55] Y. Cao and S. Liu, "Visual Servo Control for Wheeled Robot Platooning Based on Homography," pp. 628–632, 2017. doi: 10.1109/DDCLS.2017.8068145

[56] M. F. Yahya, "Image-Based Visual Servoing for Docking of an Autonomous Underwater Vehicle," pp. 1–6, 2017. doi: 10.1109/USYS.2017.8309453

[57] J. Demby, Y. Gao, A. Shafiekhani, and G. N. Desouza, "Object Detection and Pose Estimation Using CNN in Embedded Hardware for Assistive Technology," no. November, 2019. doi: 10.1109/SSCI44817.2019.9002767