# Detecting object using camera vision and inverse kinematics

*1st Ahmed Mohamed Gaber*
*School of Engineering and Applied Sciences*
*Nile University*
Giza Egypt
Email:ah.gaber@nu.edu.eg

*2nd Dr Mohamed saed*
*Wireless Intelligent Networks Center (WINC)*
*Nile University*
Giza Egypt
Email:mdarweesh@nu.edu.eg

*Abstract*—In recent years, there has been a growing interest in creating computer vision algorithms that can be used in a variety of industries. One of the most important inventions in the nineteenth century was computer vision in robotics, which allowed us to bring vision to some electronic components. It's employed in a variety of projects in the robotic industry, including automated quality control to improve manufacturing precision, productivity, and quality. The goal of this study is to learn how to create a 2DOF robotic arm in which we utilize inverse kinematics to examine structural design and show how to use various computer vision techniques to determine the location of a specific object as well as the location of the end-effector to the base frame. We used computer vision to detect object coordinates then send coordinates to the robot arm brain (Microcontroller) where inverse kinematics equations will move the robot to reach the location of object.

**Keywords** object detecting , inverse kinematics , background subtraction, color subtraction, robotic arm

## I. INTRODUCTION

Computer vision is a field of artificial intelligence (AI) that enables computers and systems to derive meaningful information from digital images, videos, and other visual inputs. If AI enables computers to think, computer vision enables them to see, observe and understand. Computer vision works much the same as human vision. Human sight has the advantage of lifetimes of context to train how to tell objects apart. Computer vision trains machines to perform these functions, but it must do it in much less time with cameras, data, and algorithms rather than retinas, optic nerves. Machine learning uses algorithmic models that enable a computer to teach itself about the context of visual data. If enough data is fed through the model, the computer will “look” at the data and teach itself to tell one image from another. Algorithms enable the machine to learn by itself, rather than someone programming it to recognize an image. As we know the computer vision is used rapidly in industrial field, it’s used in various project such as automated quality control to increase manufacturing accuracy improve productivity and produce better quality in the robotic field. Robots without visual perception capability are like blind machines developed for repetitive tasks placed in one place. Thanks to computer vision, robots are becoming intelligent to see their surroundings and move accordingly to perform various actions.

variety of issues of current interest in computer vision require the capacity to track moving objects in live streaming for purposes such as inspection, video conferencing, robot navigation, and so on. The difficulties that drive a great part of the exploration in this field are the colossal information data transfer capacity inferred by high resolution frames at high frame rates, and the yearning for real-time intuitive execution. Various innovative routine has been proposed. Nonetheless, most of these routines use complex models, for example, edges, snakes, splines, formats or computationally expensive Eigen image or condensation algorithms. Even though these methodologies are expansive in their capacities offering reliable object recognition. In addition to tracking, they are so far not able to run on full video resolution images at high frame rates. Color has been generally utilized as a real-time tracking frame works. It offers a few noteworthy points of interest over geometric signs such as computational simplicity, robustness under partial occlusion, rotation, scale and resolution changes. In the tracking framework, the color blobs are being tracked. The idea of blobs as a representation for image characteristics has a long history in computer vision and has various numerical definitions. It might be a reduced set of pixels that impart a visual property that is not imparted by the surrounding pixels. Body movement analysis is an imperative innovation which combines modern biomechanics with Computer vision. It is broadly utilized as a part of intelligent control, human machine interaction, movement analysis and different fields. Presently, systems utilized as a part of moving object detection are chiefly the frame subtraction technique, the background subtraction strategy, and the optical flow method.

The goal of this study is to use a proposed method to track moving objects in different video frames in real time. An algorithm that dissects features in video frames and gives the motion of focuses between frames is used to do video tracking. Then, using inverse kinematics to shift the location of the end effector with respect to the base frame to the desired place, extract the ROI coordinates from the video frame and transmit them to the brain of the 2DOF. There are numerous algorithms, each with its own set of strengths and weaknesses. When choose the algorithm to use, it's critical to consider the intended use. Target representation and localization, as well as filtering and information association, are two important components of a visual tracking system. Target representation and containment is usually done from the bottom up. These systems provide several capabilities for identifying moving things. The algorithm is responsible for effectively finding and tracking the target object, even if filtering and information association is typically a top-down approach that involves combining previous data about the scene or object with object dynamics and the development of various assumptions. The most encountered problem in object tracking is locating the region of interest (ROI). The region of interest is where we find the needed object in various video frames. To locate the needed object in video frames, we must first detect the object's movement using the movement estimation function, **like Centroid or Bounding box.**

## II. LITRATURE REVIEW

computer vision in robotics is playing a crucial role in making them more intelligent with the help of the right inputs. A huge amount of training data is used to train such machines and make them more intellectual to even perform crucial tasks in various fields. We are going to use python openv2 because Implementing CV through Python allows developers to automate tasks that involve visualization easily. beside bringing vision to robot we need to know the position and orientation of their end-effector with respect to their base frame. So, we will need to apply lots of math to perform this task. This known as kinematics and inverse kinematics in robotic field. Kinematics is the study of motion without considering the cause of the motion, such as forces and torques. Inverse kinematics is the use of kinematic equations to determine the motion of a robot to reach a desired position. Inverse kinematics is the use of kinematic equations to determine the motion of a robot to reach a desired position. For example, to perform automated bin picking, a robotic arm used in a manufacturing line needs precise motion from an initial position to a desired position between bins and manufacturing machines. In this paper we are going to use computer vision to bring vision to a 2DOF manipulator. 2DOF is a robot arm with 2 degree of freedom (x,y). A method to analyze the structural design and kinematic equations for a 2-DOF robotic manipulator is presented in this paper. A kinematic equations and inverse solution were established based on the analysis of the structural configuration. Then, we design and assemble the 3D mechanical parts of the robotic manipulator using a 1mm stainless steel metal bracket. With these parameters (vision and positioning robot) we can make any robot we would like to do.

## III. METHODOLOGY

### A. Object Tracking

This is the process of tracking one or more moving objects in a scene. Once the initial object has been recognized, this has usually been used to monitor real-world interactions. It's a crucial part of the self-driving cars that Uber and Tesla are planning to introduce. There are two types of object tracking: generative and discriminative. The generative method will define evident qualities and lessen the amount of time it takes to reconstruct the subject. The discriminative strategy is more effective and precise. It has become the dominant tracking method because it can discern the difference between the subject and the surroundings. Tracking-by-Detection is another name for it, and it belongs to the same category as deep learning.

### B. Image Classification and localization

Image classification allows us to categories the contents of an image. The location of a single object in an image will be specified by Image Localization. This is probably the most well-known computer vision method. How do we go about constructing computer vision algorithms that can classify images into their respective categories? This is one of the most difficult challenges to solve. To overcome the problem, there is an intriguing data-driven method. Instead of deciding how each picture category will appear on a code level, the researcher provides the computer with many images class samples for computer vision machine learning. The computer must learn about the visual appearance of the images by studying them.

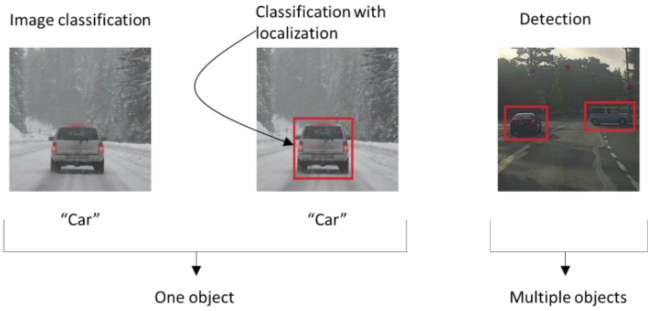



*Figure 1 image classification and localization*

### C. Semantic Segmentation

Segmentation separates the entire image into groups of pixels that may be labelled and categorized, which is an important aspect of computer vision. To be more exact, semantic segmentation tries to figure out what role each pixel in each image plays. It is not enough to detect a person or an automobile, for example. You must also be able to recognize all of the borders. We require dense pixel predictions from the models to make such demarcation.

### D. Instance Segmentation:

Instance segmentation categorizes all the various instance classes such as labeling ten cars with ten different colors. In terms of classification, there is usually the main image, and the goal is to determine what exactly the image is. However, to segment all the instances, more complex processes are required. If we have a complex scene with many overlapping objects and various backgrounds, we must classify all the objects and identify their differences, boundaries, and how they relate to one another.

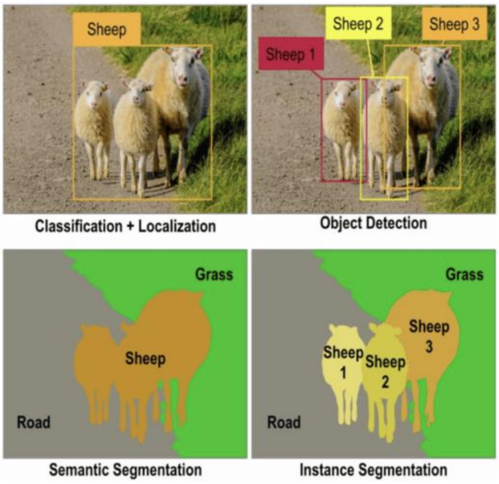



*Figure 2 image segmentation*

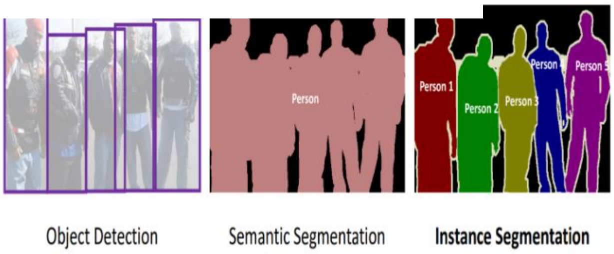



*Figure 3 semantic segmentation*

## IV. DIFFERENT TYPES OF SUBTRACTION

### A. Background Subtraction

Background subtraction (BS) is an important step in many computer vision systems since it is used to detect moving objects in a video stream without any prior knowledge of the objects. Background subtraction is a method of separating foreground and background elements, essentially creating a foreground mask (namely, a binary image containing the pixels belonging to moving objects in the scene). As the name implies, BS calculates the foreground mask by subtracting the current frame from a background model that contains the static component of the picture or, more broadly, everything that may be called background given the observed scene's features.

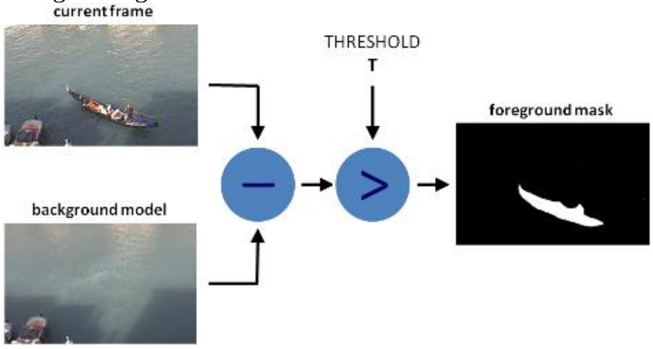



*Figure 4 background subtraction*

This method exploited the difference between the backdrop model and the current frame to detect active objects. We continue to pass each frame into the supplied function, which finds the averages of all frames. The absolute difference between the frames is then computed. *Color*

### B. Subtraction

With the help of the camera demonstrate in/on your framework, the frames are obtained-processing: It converts the color image to grey first in pre-processing since it is easier to process a grey image in a single shade rather than three hues. Grayscale photos require less processing time. The median filter is then used to remove noise from images or frames obtained from the video. The images or frames examined using the command "medfilt2" are displayed in the Image Processing Toolbox.

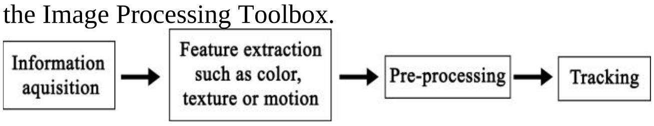



*Figure 5 color subtraction*

#### 1) Feature Extraction

Selecting the correct feature plays the major part in tracking. The feature choice is nearly identified with the object representation. Different features needed for tracking are color, edges, optical flow, and surface or texture. In the proposed approach, we track the target object utilizing the color feature, particularly red, green, and blue shade, in same way it can track the red, green, and blue color objects in the video. After tracking the target object using color feature, we are going to track object using motion. For motion detection and tracking we are using Frame difference technique and optical flow algorithm.

*C. Tracking:*

The tracking of the target objects is based on the object's location parameters, such as Bounding box, Area, Centroid, and so on. To track, the bounding box property is used. As the object goes through different places in the movie, the Bounding box moves along with it, resulting in various estimates of area attributes. As a result, the goal of tracking objects in real time utilizing color and motion has been met.

In the form of a flowchart, the suggested algorithm for object detection and tracking utilizing color feature is shown

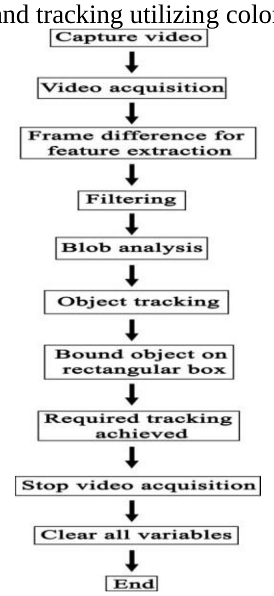



*Figure 7 tracking*

## V. INVERSE KINEMATICS

In our two-joint robotic arm, given the angles of the joints, the kinematics equations give the location of the tip of the arm. Inverse kinematics refers to the reverse process. Given a desired location for the tip of the robotic arm, what should the angles of the joints be to locate the tip of the arm at the desired location. There is usually more than one solution. This is a typical problem in robotics that needs to be solved to control a robotic arm to perform tasks it is designated to do. In a 2-dimensional input space, with a two-joint robotic arm and given the desired coordinate, the problem reduces to finding the two angles involved. The first angle is between the first arm and the ground (or whatever it is attached to). The second angle is between the first arm and the second arm.

To find inverse kinematics: These approaches are mainly divided into two types.

1. Analytical solutions.
2. Approximate Solutions.

*1) Analytical Solutions:*

In Analytic solution to an inverse kinematics problem, we have a closed-form expression which gives you the inverse kinematics (joint variables) as a function of the end-effector pose. Which means we have a definite equation for each joint parameter. We just must put the known values in the equations (like end-effector's target pose, robot link lengths) to get the joint parameters required to achieve that

$$q_1=f_1(x)$$

$$q_2=f_2(x)$$

....

$$q_n=f_n(x)$$

*Figure 6 joint parameters*

q's are the joint parameters
f(x) is the equation in terms of known values (position and orientation of end effector, link lengths)
Few approaches in Analytical Solutions are

Algebraic Approach
In this approach, we use the equations derived by equating the give Transformation matrix [target position and orientation] and the obtained Forward kinematics matrix of the robot. Using these equations, we will calculate the joint parameters. q1-q2. qn.

*2) Approximate Solutions*

Approximate solutions are another name for approximate solutions. We do more than just input numbers into an expression when we numerically compute the joint angles corresponding to an end-effector posture. Approximate solutions are frequently based on iterative optimization, in which the target position is approached by moving closer to it with each iteration.

There are a few approaches to approximate solutions.

- **Jacobian Inverse technique**
- **Cyclic Co-ordinate descent**

In this paper we used the analytical solution to find our coordinates of the end effector with respect to base frame. This is our kinematics diagram for our manipulator.

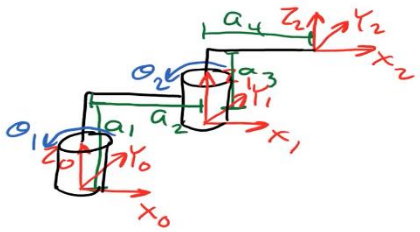
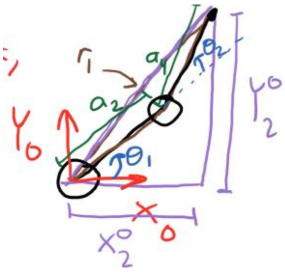



*Figure 8 kinematics diagram*

## VI. EQUATIONS

Now we will draw triangle that will show us the position of end effector.
Theta1: angle between base frame and ground,
Theta2: angle between base frame and second frame.
Y0-2: y position of end effector in base frame. X0-2: X position of end effector in the base frame

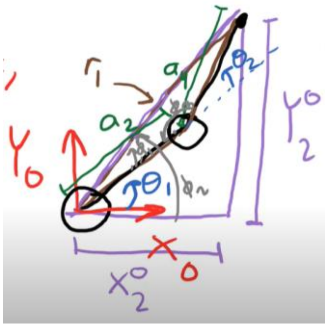


To solve this triangle first we will use Pythagorean theorem.

$r1^2 = (x0-2)^2 + (y0-2)^2$

$r1 = \sqrt{(x0-2)^2 + (y0-2)^2}$ (1)

Then we will add $\phi1$ and $\phi2$ on the top view of the robot arm.



2)$\Theta_1 = \phi1 - \phi2$ (2)
3)Finding $\phi_2$

$\tan\phi_2 = y_{0-2} / x_{0-2}$
$\phi_2 = \tan\phi2^{-1}(y0-2 / x0-2)$ (3)
4) using cos law to find $\phi1$.
$a4^2 = a2^2 + r1^2 - 2 * a2 * r1 * \cos\phi1$
$\phi1 = \cos^{-1}(a4^2 - a2^2 - r1^2 / -2 * a2 * r1)$ (4)

5) using cos law to find $\phi3$
$\Phi3 = \cos^{-1}(r1^2 - a2^2 - a4^2 / -2 * a2 * a4)$ (5)

6) finding $\Theta2$
$\Theta2 = 180 - \phi3$ (6)

## VII. BUILDING THE ROBOT ARM

I have designed and built the 2DOF manipulator using metal brackets 1mm stainless steel sheet I cut it with angle grinder. I have used 13kg servo motor for controlling joints of the manipulator. The brain of our robot is Arduino uno where it’s connected via serial communication with the laptop to receive the coordinates from python code via PY Serial library.

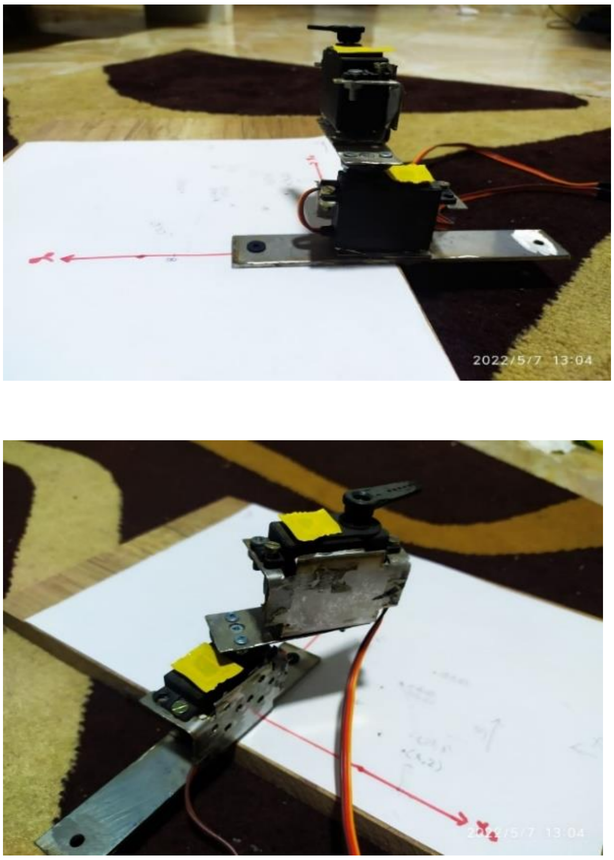


*Figure 9 image of the robotic arm*

## VIII. RESULTS

The algorithms or approach described in this paper has been used for object detection and tracking. The output obtained using these algorithms or methods are shown below.

Fig. 9 shows the detection and tracking of blue color object and shows its centroid value in terms of x and y.

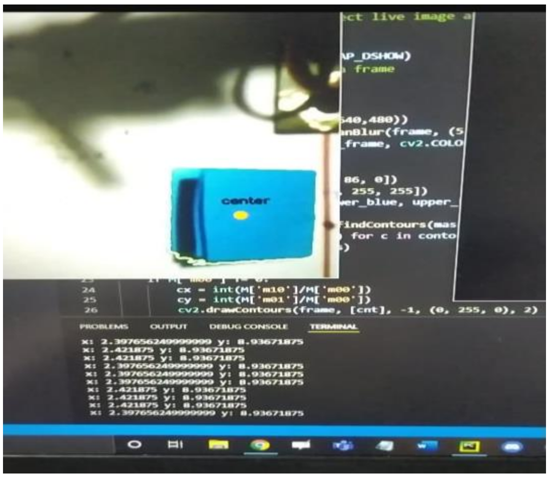



*Figure 10 detecting of blue color obj*

*Description*

In the above figure we used the blue color subtraction method. Where the blue color only is detected in the image and using contours, we detect blue obj in the image, we took the largest contour in the image means detecting largest blue color. then draw a green boundary box around detected obj. Then detecting position of its center to be sent to Arduino to move manipulator using inverse kinematics.

## IX. CONCLUSION

Adding computer vision to the field of robotics has dramatically changes the field of industry and destroys the boundaries of robotic field. Using the background subtraction method was more accurate than using the color subtraction method to find an object coordinates. During testing we found than Any noise can affect color subtraction method accuracy. Background subtraction Technique is not limited to a specific color so it can used in various projects in different fields than color subtraction method.

## X. REFERENCES


[1] Kinematic analysis and simulation of a 3-DOF robotic manipulator. IEEE Xplore. (n.d.). Retrieved March 30, 2022, from https://ieeexplore.ieee.org/document/7977291#:~:text=Abstract%3A-,Three%20degrees%20of%20freedom%20(3%2DDOF)%20robotic%20manipulator%20is,is%20presented%20in%20this%20paper.

[2] Google. (n.d.). Computer vision. Google Books. Retrieved March 30, 2022, from https://books.google.com.eg/books?hl=en&lr=&id=PmrICLzHutgC&oi=fnd&pg=PA1&dq=how%2Bto%2Blearn%2Bcomputer%2Bvision&ots=cTMWWnK51u&sig=Em4dEyY-9ciz5LiYOivaIlveYk&redir_esc=y#v=onepage&q=how%20to%20learn%20computer%20vision&f=false

[3] Tolani, D., Goswami, A., & Badler, N. I. (2002, May 25). Real-time inverse kinematics techniques for anthropomorphic limbs. Graphical Models. Retrieved March 30, 2022, from https://www.sciencedirect.com/science/article/abs/pii/S1524070300905289

[4] M. H. Ali, Fadhan Hafiz, A. A Shafie, "Motion Detection Techniques using Optical Flow'', *International Scholarly and Scientific Research and Innovation*, vol. 3, no.8, pp. 503-508, Sept 2009.

[5] Wei Shuigen, Chen Zhen, Dong Hua,"Motion Detection Based on Temporal Difference Method and Optical Flow field'', *presented at International Symposium on Electronic Commerce and Security*, Nanchang, China, May 22-24, 2009, pp. 85-89.

[6] K. Suganya Devi, N. Malmurugan, M. Manikandan, "Object Motion Detection in Video Frames Using Background Frame Matching'', *International Journal of Computer Trends and Technology*, vol.4, no. 6, pp. 1928-1931, June 2013.

[7] Nan Lu, Jihong Wang, Q.H Wu, Li Yang, "An Improved Motion Detection Method for Real-Time Surveillance'', *International Journal of Computer Science*, vol. 35, no. 1, Feb 2008.

[8] K.Kavitha, A.Tejaswini, "Background Detection and Subtraction for Image Sequences in Video'', *International Journal of Computer Science and Information Technologies*, vol. 3, no. 5, pp. 5223-5226, Oct 2012.