**Computer Vision 3D: Into the Unknown Dimension**

An ilustrative tutorial on how to convert 2D information into 3D by using simple Computer Vision and spatial geometry.

[[Andrés Matesanz](https://medium.com/@matesanz.cuadrado?source=post_page-----b742ce7f791f--------------------------------)](https://medium.com/@matesanz.cuadrado?source=post_page-----b742ce7f791f--------------------------------)

[Andrés Matesanz](https://medium.com/@matesanz.cuadrado?source=post_page-----b742ce7f791f--------------------------------)

·

[Follow](https://medium.com/m/signin?actionUrl=https%3A%2F%2Fmedium.com%2F_%2Fsubscribe%2Fuser%2Ff120f1e78bd7&operation=register&redirect=https%3A%2F%2Fmedium.com%2F%40matesanz.cuadrado%2Fcomputer-vision-3d-into-the-unknown-dimension-b742ce7f791f&user=Andr%C3%A9s+Matesanz&userId=f120f1e78bd7&source=post_page-f120f1e78bd7----b742ce7f791f---------------------post_header-----------)

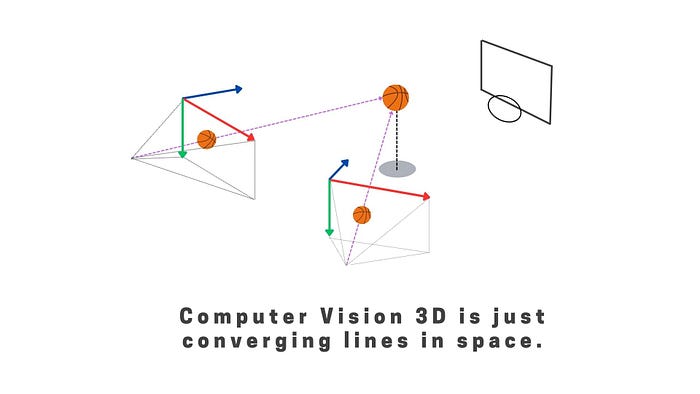
11 min read

·

Nov 21, 2023

263

**Computer vision aims to provide computers with a form of sight similar to human vision**. Many projects in this field focus on extracting 2D information from images, such as identifying objects or determining human poses. **However, our perception of the world is not in pixels but in meters (or feet)**. 3D computer vision enables the measurement of distances, velocities, accelerations, and more. In this post, I’ll guide you through the basics of converting 2D information into 3D.



Source: by Author

At **[Sngular](https://www.sngular.com/" \t "_blank)** we have a lot of **experience developing Computer Vision products**(2D and 3D 😊) **.** If you are intereseted in our work, please check more of our articles:

**[Rebuild Your Home in 3D with Gaussian Splatting](https://medium.com/@adrian.milla/rebuild-your-home-in-3d-with-gaussian-splatting-f6d7222717fb?source=post_page-----b742ce7f791f--------------------------------" \t "_blank)**

[Artificial Intelligence is probably one of the major fields in our days, it is being used for basically anything from…](https://medium.com/@adrian.milla/rebuild-your-home-in-3d-with-gaussian-splatting-f6d7222717fb?source=post_page-----b742ce7f791f--------------------------------" \t "_blank)

[medium.com](https://medium.com/@adrian.milla/rebuild-your-home-in-3d-with-gaussian-splatting-f6d7222717fb?source=post_page-----b742ce7f791f--------------------------------" \t "_blank)

**[AI Wizardry: Transforming a Photo into Endless Perspectives](https://medium.com/@adrian.milla/ai-wizardry-transforming-a-photo-into-endless-perspectives-ed186ec8b4c9?source=post_page-----b742ce7f791f--------------------------------" \t "_blank)**

[In the realm of artificial intelligence, we often witness astounding advancements that reshape the way we interact with…](https://medium.com/@adrian.milla/ai-wizardry-transforming-a-photo-into-endless-perspectives-ed186ec8b4c9?source=post_page-----b742ce7f791f--------------------------------" \t "_blank)

[medium.com](https://medium.com/@adrian.milla/ai-wizardry-transforming-a-photo-into-endless-perspectives-ed186ec8b4c9?source=post_page-----b742ce7f791f--------------------------------" \t "_blank)

**[Revolutionizing Sports Advertisement with Generative AI.](https://medium.com/@matesanz.cuadrado/revolutionizing-sports-advertisement-with-generative-ai-2bd1c1134c6e?source=post_page-----b742ce7f791f--------------------------------" \t "_blank)**

[A guide on how to use generative AI to create custom user advertisements in Football.](https://medium.com/@matesanz.cuadrado/revolutionizing-sports-advertisement-with-generative-ai-2bd1c1134c6e?source=post_page-----b742ce7f791f--------------------------------" \t "_blank)

[medium.com](https://medium.com/@matesanz.cuadrado/revolutionizing-sports-advertisement-with-generative-ai-2bd1c1134c6e?source=post_page-----b742ce7f791f--------------------------------" \t "_blank)

**[Less Pain, More Gain.](https://medium.com/@loboateresa/less-pain-more-gain-3737fe287079?source=post_page-----b742ce7f791f--------------------------------" \t "_blank)**

[Lift Weights With Your AI Trainer — A Tutorial](https://medium.com/@loboateresa/less-pain-more-gain-3737fe287079?source=post_page-----b742ce7f791f--------------------------------" \t "_blank)

[medium.com](https://medium.com/@loboateresa/less-pain-more-gain-3737fe287079?source=post_page-----b742ce7f791f--------------------------------" \t "_blank)

Good enought! Let’s start!

**A Quick jump into Human Vision**

Let’s begin by a simple question**:** **if Mike Wazowski wanted to join your basket team, at a first glimpse, would you think he would perform good or bad?**

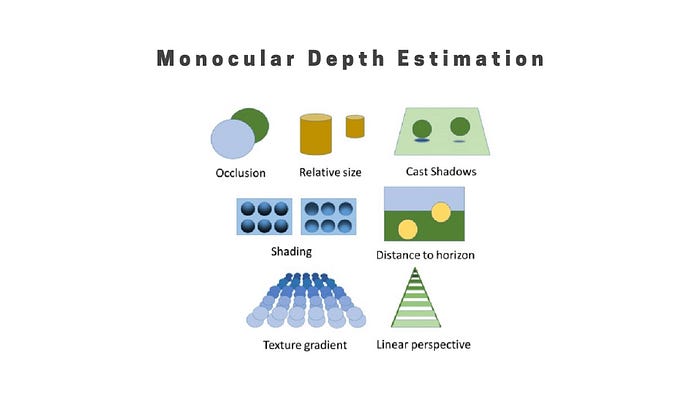


Would Mike Wazowski be a good basket player? (Source: [Bing Chat](https://www.bing.com/search?q=Bing+AI&showconv=1))

Basket quality relies in the ability of the player to percieve depth. So, to answer this question, we first need to understand how would Mike perceive depth with his only eye.

**Monocular Depth Estimation**

**Monocular depth estimation** **relies on a multitude of visual cues** embedded in a single 2D image, enabling the inference of depth relationships within a scene. **Occlusion** plays a pivotal role, as objects partially obstructed by others are typically situated farther away. The **relative size** of objects offers another cue, where closer entities appear larger, while those in the distance seem smaller. **Texture gradient** is a perceptual phenomenon wherein the density and size of textures vary with an object’s distance, providing further depth-related information.



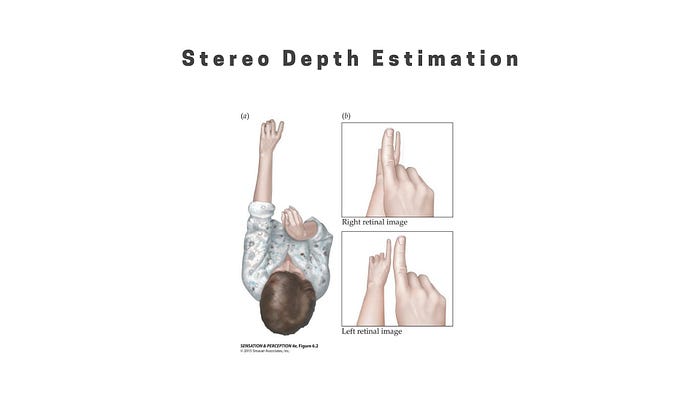
Monocular depth estimation relies on a multitude of visual cues (Source: [Depth Perception of Surgeons in Minimally Invasive Surgery](https://www.researchgate.net/figure/Illustration-of-the-basic-monocular-depth-cues_fig4_299401615))

**Linear perspective**, dictated by the convergence of parallel lines as they extend into the distance, contributes significantly to depth estimation. **Shading and lighting**, including shadows and highlights, aid in deducing the three-dimensional shapes of objects within the scene. **Motion parallax,** observed as differing rates of movement for objects at various depths as an observer moves, provides a dynamic depth cue (while driving, mountains move slower than road signs). Together with **aerial perspective,** which manifests as changes in saturation, blurring, and tinting for distant objects due to atmospheric effects, these cues form a comprehensive set for monocular depth perception. Focus and defocus, depth from motion, and considerations of familiar size further enrich the complexity of depth estimation in both human vision and computer vision systems.

Even if **Mike Wazowski** possessed monocular depth estimation capabilities, he would still **lack the robustness of stereo vision** — a powerful depth perception mechanism in humans. Stereo vision relies on having two eyes, providing slightly different perspectives of a scene, which **enables more accurate and reliable depth estimation** than monocular cues alone. Humans are adept at leveraging this binocular disparity to precisely gauge distances.

**Stereo Depth Estimation**

In stereo vision, each eye captures a slightly different view of the world, creating a binocular disparity. **The brain then combines these disparate images into a single, three-dimensional perceptual experience**. This process, known as **stereopsis**, occurs in the visual cortex where the brain analyzes the differences between the images received by each eye. The greater the difference, or disparity, between the images, the closer the object is perceived to be. This dynamic and real-time interplay between the eyes and the brain allows humans to excel in tasks requiring accurate depth perception, such as judging distances, grasping objects, and navigating the environment with precision. While monocular depth cues are valuable, stereo vision significantly enhances the richness and accuracy of our perception of the three-dimensional world.



Each eye captures a slightly different view of the world, creating a binocular disparity. (Source: [Princeton Depth Perception, Part II](https://pillowlab.princeton.edu/teaching/sp2017/slides/Lec13_Depth_Chap6b.pdf))

Similarly to human stereopsis, computers employ a triangulation approach to estimate the three-dimensional positions of objects in space. This method involves analyzing the disparities or differences in the perspectives of multiple cameras or sensors, mimicking the way our two eyes work together. By capturing slightly offset views of a scene, computational systems can calculate the disparities and use trigonometry to determine the distances to various points in the environment. This process, often utilized in computer vision and depth sensing technologies, enables machines to create accurate 3D reconstructions of scenes and objects. While lacking the biological intricacies of human vision, this computational triangulation approach harnesses the power of stereo vision principles to enhance the depth perception capabilities of machines in various applications, from robotics to augmented reality.

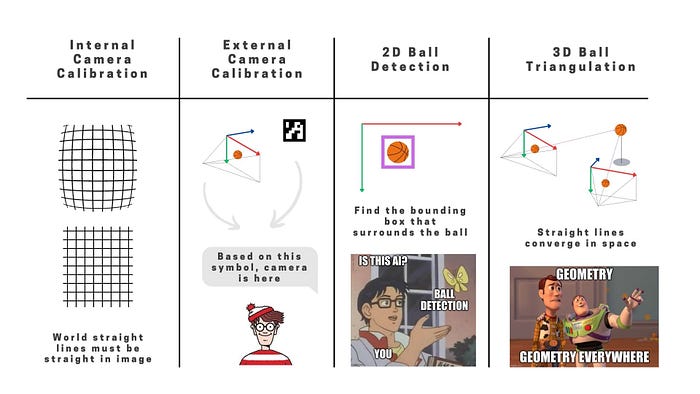


Computers employ a triangulation approach to estimate the three-dimensional positions of objects in space. (Source: [IDS Imaging](https://en.ids-imaging.com/technical-articles-details/whitepaper-depth-information-3d-images.html))

But, what are the steps to give computers this 3D perception ability? Let’s dive in.

**How to: Computer Vision 3D**

Computer vision in 3D involves a sequence of crucial steps to transform 2D images into a comprehensive understanding of the three-dimensional world:



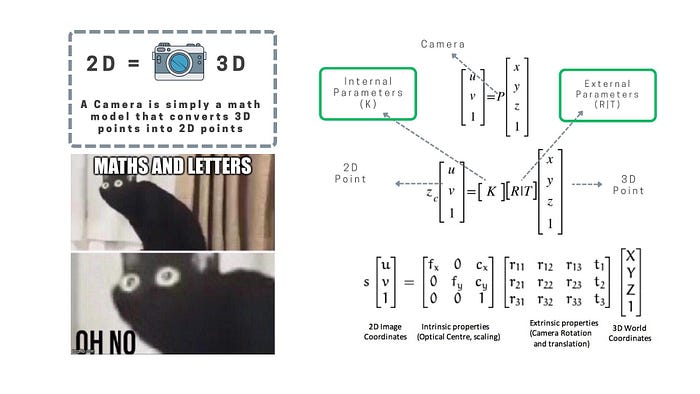
the 4 steps involved in 3D triangulation. (Source: by Author)

* The first step, **internal camera calibration**, focuses on fine-tuning the intrinsic parameters of the camera, such as focal length and lens distortion, to ensure accurate and consistent imaging.
* **External camera calibration** follows, aligning multiple cameras in a system and establishing their spatial relationships, a key process for accurate triangulation.
* **Object detection** comes next, where computer algorithms identify and locate objects within the images.
* Finally we have **Object triangulation**. Leveraging the principles of stereo vision, this step calculates the precise 3D positions of objects by using a geometric technique, to determine where the rays of sight from each camera intersect in three-dimensional space. The point of convergence represents the precise 3D position of the object.

Together, these four steps form the backbone of computer vision in 3D, enabling machines to perceive and interact with the world in a spatially aware manner.

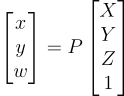
**What is a camera?**

Before talking about camera calibration we need to understand **what a camera is**, at least mathematically.



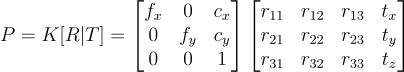
Camera (P) decomposed into internal (K) and external parameters (R|T). (Source: by Author)

A camera, mathematically speaking, is a crucial tool in computer vision and image processing. At its core, it can be represented a**s a mathematical model that transforms three-dimensional points in space into two-dimensional points** on an image plane. This transformation is encapsulated by the projection matrix (P), where a 3D point (X, Y, Z, 1) is projected onto the image plane as (x, y, 1) using the equation:



Camera (P) as a “conversor” (matrix) from 3D points (world) into 2D points (image).

Breaking down the camera model, we encounter two sets of parameters: internal and external. **Internal parameters** are denoted by the calibration matrix (K) and may include focal length (fx, fy), optical center (cx, cy), and distortion parameters (k\_1, k\_2..). **External parameters** (R|T) define the camera’s position and orientation in space. Here, R represents the rotation matrix, and T represents the translation vector. The projection matrix (P) can then be decomposed as:

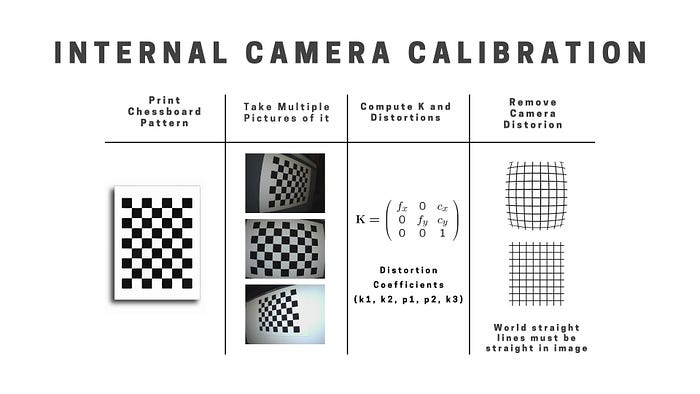


Camera (P) decomposed into internal (K) and external parameters (R|T).

*One common method for internal calibration is to use a****printed chessboard pattern****.*

**Internal Camera Calibration**

OK but, **how do we obtain this internal parameters in real life?** One common method for internal calibration is to use a printed chessboard pattern. The process typically involves the following steps:



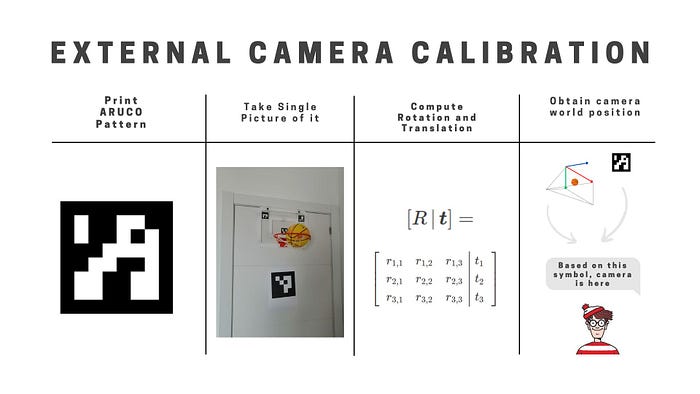
Internal Camera Calibration Steps. (Source: by Author)

* **Print a chessboard pattern** with well-defined square corners and edges. The known geometry of the chessboard provides reference points for calibration.
* **Take multiple images of the chessboard**from different angles and orientations using the camera that needs calibration. Ensure that the entire chessboard is visible in each image.
* **Apply calibration algorithms**, such as Zhang’s method or OpenCV’s calibration functions, to the detected chessboard corners. These algorithms use the known 3D coordinates of the chessboard corners and their 2D image coordinates to compute the intrinsic calibration matrix (K), which includes focal length, optical center, and lens distortion parameters.

This calibrated camera model allows for accurate mapping of three-dimensional points in the world onto the two-dimensional image plane.

**External Camera Calibration**

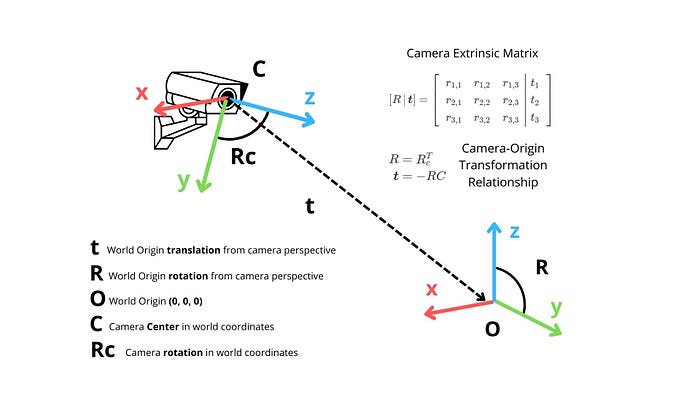
External camera calibration is the process of determining the position and orientation of a camera in the three-dimensional world relative to a known reference frame. This involves finding the external parameters, typically represented by the rotation matrix (R) and translation vector (T), which describe the camera’s pose in the global coordinate system.



External Camera Calibration Steps. (Source: by Author)

And **how do we obtain this internal parameters in real life?**One practical and widely-used method for external camera calibration involves **using ArUco markers.**

* **Place ArUco markers with known dimensions in the scene**. ArUco markers are black-and-white square patterns with a unique identifier. Ensure that the markers are visible from multiple viewpoints.
* Use the camera to **capture images of the** scene with the **ArUco markers** from different angles and distances. Make sure that each marker is clearly visible in the images.
* Utilize computer vision libraries, such as OpenCV, to **detect and identify the ArUco markers** in each image. The detected markers serve as reference points for calculating the camera’s external parameters.
* With the known 3D coordinates of the ArUco markers and their corresponding 2D image coordinates, **apply camera calibration algorithms** to compute the rotation matrix (R) and translation vector (T) that define the camera’s position and orientation in the global coordinate system.

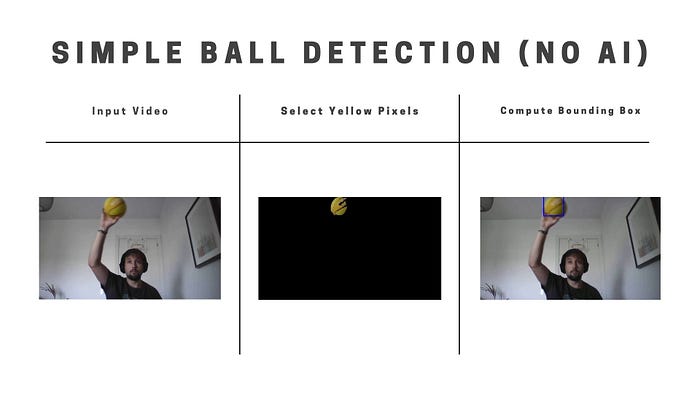


Camera Rotation and Position from world origin. (Source: by Author).

Knowing both the internal and external parameters of a camera provides a comprehensive understanding of its intrinsic characteristics and its **position and orientation in the three-dimensional world**. With this information, it becomes possible to compute straight vectors in 3D space, allowing for accurate mapping of points from the camera’s perspective to the global coordinate system.

**Object Detection**

Object detection is crucial for triangulation purposes in computer vision because it provides the essential information needed to establish **correspondence between the 2D image space and the 3D world.**



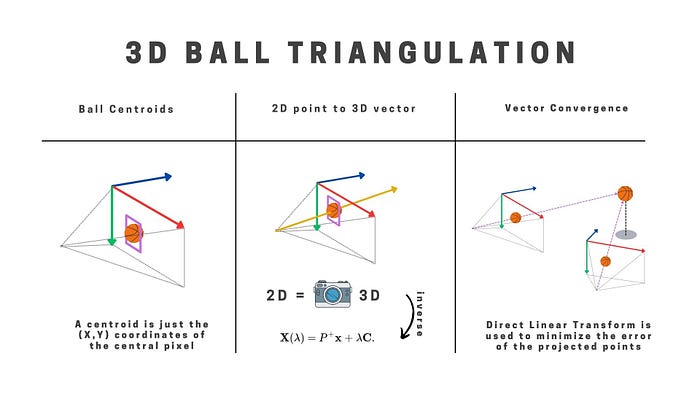
Steps to detect a yellow baskeball using a classical computer vision approach. (Source: by Author)

In object detection, various approaches can be employed, ranging from advanced AI models like YOLO (You Only Look Once) to classical computer vision methods. We opted for a classical computer vision approach by masking yellow pixels to detect a basketball, the steps needed were:

* **Color Segmentation**: In the first stage, convert the image to the HSV color space. This conversion is crucial as it separates intensity from color information, making it easier to focus on the hue, saturation, and value components. Define the yellow color range in the HSV space to isolate the pixels corresponding to the yellow color of the basketball. Create a binary mask by thresholding the image based on this defined color range.
* Moving to the second stage, **identify contours within the binary mask**. Contours represent continuous regions of yellow color in the image. Following contour detection, filter the identified contours based on various criteria. This may involve excluding contours that are too small or have irregular aspect ratios, ensuring that only relevant regions corresponding to the basketball are considered.
* In the final stage, **calculate bounding boxes** for the filtered contours. These bounding boxes represent the spatial extent of the detected yellow basketball regions. Each bounding box is characterized by its coordinates (x, y) and dimensions (width, height). Optionally, draw these bounding boxes on the original image to visually highlight the regions identified as containing a yellow basketball.

**Object Triangulation**

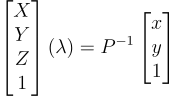
Triangulating an object in 3D involves using the inverse of a camera model and the Direct Linear Transform (DLT) algorithm.



Steps to triangulate in 3D space a baskeball. (Source: by Author)

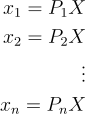
Here’s a step-by-step explanation:

* At the begining we saw that a camera is a math model that converts 3D points into 2D points.**If we invert the process, we can get the 3D points out of 2D points, well, at least the vectors pointing to our object. Inverse camera projection** is the process of transforming 2D image coordinates back into 3D world coordinates. The inverse of the camera matrix, is utilized for this purpose. It can be computed as **the pseudo-inverse of the camera matrix**, as the camera matrix is typically not square. The inverse camera projection is represented as:



Pseudo-inverse of the camera matrix to compute vectors pointing to objects.

* The **Direct Linear Transform**(DLT) **algorithm** is applied to triangulate 3D points from corresponding 2D points in multiple camera views. For each view, the equation (x = PX) is used, where (x) is the 2D image coordinates, (P) is the camera matrix, and (X) is the 3D point. The system of equations can be expressed as:



System of equations to solve DLT and triangulate multiple points in 3D space.

By applying the inverse camera model and the DLT algorithm for each camera view, you can triangulate the 3D coordinates of a point in space. This process is fundamental for tasks such as 3D reconstruction and scene understanding in computer vision applications.

**Conclussion**

In this article, we explored the crucial steps of **internal and external camera calibration, object detection, and object triangulation**, collectively forming the foundation for comprehensive 3D perception in computer vision.

**Internal camera calibration** fine-tunes intrinsic parameters, ensuring accurate imaging. **External camera calibration** aligns multiple cameras and establishes spatial relationships, essential for accurate object triangulation. **Object detection,** a pivotal step, identifies and locates objects within images, setting the stage for the final critical process — **object triangulation**. Leveraging calibrated cameras, triangulation calculates precise 3D positions of objects.

Together, these steps e**nable machines to perceive and interact with the three-dimensional world accurately**, with applications ranging from robotics and autonomous navigation to augmented reality. This holistic approach highlights the interdisciplinary nature of computer vision, where precise calibration, effective detection, and accurate triangulation converge to unlock the potential for machines to comprehend and navigate our spatial reality.

*👋 Hey! Did you liked this article? You can connect with me on*[***LinkedIn***](https://www.linkedin.com/in/aimatesanz/)*or at my*[***Personal Web Page***](https://matesanz.github.io/)*and, please,****don’t forget to leave some claps!****👏 😃 💙*