# Depth Estimation

## Introduction:

Depth estimation in computer vision is the process of determining the depth of different objects present in a captured image and visualizing the 3-D representation from a 2-D image. There are several methods for performing depth estimation including monochrome cameras, stereo cameras, and RGBD cameras.

## Applications:

- object detection
- 3D mapping
- parallax simulation
- refocus
- autonomous navigation
- robotics simulation and manipulation
- virtual reality (VR)

Now, we will explore the different approaches to estimate depth and obtain a 3-D representation and their different techniques:

### 1. Monochrome Cameras:

Monochrome cameras, also known as black and white cameras, use a single lens and image sensor to capture images. This method estimates depth by analyzing the different gray color intensities or shades of pixels between an object and its background. However, this technique has limitations in its ability to provide accurate depth information.

Here are two commonly used techniques:

#### 1. Structure from motion (SFM):

SFM estimates the 3D structure of an image and the camera motion simultaneously, by analyzing the movement of the camera and the positions of the features in the images.

Here is how SFM works:

1. Feature Detection and Matching: The first step in SFM is to detect and track feature points such as corners or blobs in each frame of the image sequence. Then establishing correspondences between the feature points across different frames. This is done by matching the detected features based on their descriptors, which are compact representations of the local image patch around each feature point.
2. Camera Pose Estimation: With the correspondences established, the next goal is to estimate the camera's pose for each frame. This involves determining the position and orientation of the camera relative to the scene.
3. Triangulation: After estimating the camera poses, the next step is to triangulate the 3D positions of the feature points. Triangulation involves finding the 3D coordinates of a feature point by intersecting the corresponding rays projected from different camera viewpoints. This is done by solving a system of equations that relates the 2D image coordinates of the feature point with the camera poses.
4. Depth Estimation: With the 3D point positions, the depth of the scene can be estimated. The depth of a point can be calculated as the distance between the camera and the 3D point. This depth information provides an estimate of the scene's geometry and the relative distances between objects.

In conclusion, SFM is a powerful technique and has many applications in computer vision, such as 3D reconstruction, augmented reality, and robotics. However, it can be computationally expensive and requires careful parameter tuning to achieve good results.

**2. Monocular (SLAM):**

Monocular Simultaneous Localization and Mapping (SLAM) is a technique that combines visual odometry and mapping to estimate the camera's motion and simultaneously build a map of the environment using a single monocular camera.

Here is how monocular SLAM works:

1. Feature Detection and Tracking: The first step in monocular SLAM is to detect and track feature points in the image sequence captured by the monocular camera. These feature points are typically distinctive keypoints such as corners or blobs.
2. Camera Pose Estimation (Visual Odometry): Once the feature points are detected and tracked across frames, the camera's pose is estimated. Visual odometry techniques analyze the changes in the position of these feature points to calculate the camera's motion. By comparing the feature points' positions in successive frames, the camera's translation and rotation can be estimated.
3. Keyframe Selection: To build a map of the environment, keyframes need to be selected from the sequence of frames. Keyframes are frames that are chosen at regular intervals or when significant camera motion is detected.
4. Map Initialization: Once keyframes are selected, the map initialization process begins. The initial map is created by triangulating the 3D positions of the feature points observed in the keyframes. This is achieved by solving the perspective n point (PnP) problem, which relates the 2D image coordinates of the feature points with their corresponding 3D positions.
5. Loop Closure Detection: Monocular SLAM systems also incorporate loop closure detection to handle revisited or previously observed areas. Loop closures occur when the camera revisits a location, allowing the system to recognize and close the loop by detecting and connecting the current frame to a previously visited keyframe. Loop closures help improve the accuracy of the map and reduce drift in the camera's pose estimation.
6. Map Maintenance and Localization: Once the map is built, it can be used for localization. The camera can compare the features observed in the current frame with the map's feature points to estimate its position and orientation in real-time. This localization step helps in maintaining the camera's pose estimation and assists in navigation or augmented reality applications.

By performing these steps, monocular SLAM systems can estimate the camera's motion, build a map of the environment, and localize the camera in real-time using a single monocular camera. Monocular SLAM has applications in robotics, augmented reality, autonomous vehicles.

## 2. stereo cameras:

Stereo camera is type of camera with two or more cameras arranged side by side with image sensor for each camera lens. This camera simulates the human binocular vision and gives the ability to capture 3-D images by capturing images from different viewpoints.

Here are two commonly used techniques:

**1. Stereo correspondence (stereo matching):**

Stereo correspondence also known as stereo matching, is a technique that estimates depth by finding correspondences between pixels in the left and right images captured by the stereo camera.
Here is how stereo correspondence works:

1. Image Rectification: The left and right images are rectified to align corresponding epipolar lines (straight line of intersection of the epipolar plane with the image plane). This simplifies the matching process by transforming the stereo disparity problem into a 1D search along the epipolar lines.
2. Matching: For each feature point or pixel in the left image, a search is performed along the corresponding epipolar line in the right image to find the best matching point.
3. Disparity Estimation: The disparity, which represents the horizontal shift between the corresponding points in the left and right images, is calculated based on the matched points. The disparity value is inversely proportional to the depth, so a larger disparity indicates a closer object.

4. Depth Calculation: With the disparity values obtained, the depth of each pixel can be estimated using triangulation. By knowing the distance between the two camera viewpoints and the baseline distance between the cameras, the depth of a point can be calculated using the triangulation equation.

#### 2. Dense depth estimation:

Dense depth estimation is a technique that estimates depth for every pixel in the stereo images, providing a dense depth map of the scene. It leverages the concept of matching and disparity estimation but operates at a pixel wise level rather than detecting individual feature points.

Here is how dense depth estimation works:

1. Cost Calculation: A cost volume is constructed by computing the matching cost for each pixel in the left image with its corresponding search window in the right image. Matching costs can be calculated using measures like absolute differences or normalized cross-correlation.
2. Disparity Optimization: The cost volume is then optimized to find the best disparity value for each pixel. This can be achieved through techniques like disparity refinement, cost aggregation, or optimization algorithms such as graph cuts or belief propagation.
3. Disparity to Depth Conversion: The estimated disparities are converted to depth values using the triangulation equation, similar to the stereo correspondence technique. The resulting depth map provides depth information for each pixel in the scene.

Dense depth estimation techniques offer higher resolution and denser depth maps compared to sparse stereo correspondence techniques, but they are typically more computationally intensive.

## 3. RGBD cameras:

An RGBD camera, also known as a depth camera or a 3D camera, is a type of camera that captures both color (RGB) and depth (D) information of the scene simultaneously. Unlike a stereo camera that relies on image disparity to estimate depth, an RGBD camera directly provides depth measurements for each pixel in the captured image.

Here are two commonly used techniques:

#### 1. Time of Flight (ToF):

RGBD cameras based on the Time-of-Flight principle emit modulated light signals, such as infrared (IR) light, and measure the time it takes for the light to travel to the scene and bounce back to the camera. The depth information is derived from the time it takes for the light to make the round trip.

Here is how time of flight works:

1. Light Emission: The RGBD camera emits a pulsed light signal, usually in the form of infrared light, towards the scene.
2. Light Reflection and Capture: The emitted light reflects off the objects in the scene and returns to the camera.
3. Time Measurement: The RGBD camera measures the time it takes for the emitted light signal to travel to the scene and bounce back. This measurement is typically achieved using sensors that can accurately measure the time of flight of the light, such as photodiodes or specialized CMOS sensors.
4. Depth Calculation: The depth information is derived from the measured time of flight. By knowing the speed of light, the round-trip time can be converted into distance, providing depth measurements for each pixel in the captured image.

#### 2. Structured Light:

RGBD cameras based on structured light projection employ the projection of a known pattern onto the scene and analyze the deformation of the pattern to estimate depth.

Here is how structured light works:

1. Pattern Projection: The RGBD camera projects a known pattern, such as a grid or a set of coded light patterns, onto the scene.
2. Deformation Analysis: The camera captures the deformed pattern as it appears on the objects in the scene. The deformation of the pattern provides information about the geometry and depth of the objects.
3. Pattern Decoding: The RGBD camera analyzes the captured deformed pattern to determine correspondences between the projected pattern and the captured pattern. This decoding process involves identifying features or points in the pattern and matching them between the projected and captured images.
4. Depth Calculation: The depth information is derived from the correspondences between the projected and captured patterns. By knowing the geometry of the projected and the correspondences the depth of each pixel can be calculated.

## Resources:

- https://huggingface.co/tasks/depth-estimation
- https://stanford.edu/class/ee367/Winter2017/Project proposals/weinberger_wang_ee367_win17_proposal.pdf
- https://www.adorama.com/alc/monochrome-cameras/#:~:text=Simply put%2C a monochrome camera,colors from across the spectrum
- https://www.mathworks.com/help/vision/ug/structure-from-motion.html
- https://www.hindawi.com/journals/js/2017/6842173/?utm_source=google&utm_medium=cpc&utm_campaign=HDW_MRKT_GBL_SUB_ADWO_PAI_DYNA_SPEC_X_X0000_Feb
- https://en.m.wikipedia.org/wiki/Stereo_camera