Course 2: Advanced Robotics

Chapter1 : Kinematics and Dynamics

•**Forward Kinematics**:

•Understand how to compute the position of a robotic arm based on its joint angles.

•Use transformation matrices to model 2D robotic arms.

•**Inverse Kinematics**:

•Calculate the joint angles required to move the arm to a specific position.

•Solve simple systems of equations manually or using MATLAB.

2/ Introduction to ROS (Robot Operating System)

- **Installation**:
- Set up ROS on Ubuntu (recommend ROS2).
- Understand the architecture: nodes, topics, services, and messages.
- **First Project**:
- Control a simulated robotic arm in Gazebo or RViz.

3/ **Path Planning**

- **Algorithms**:
- Learn pathfinding algorithms like A* and Dijkstra.
- Understand how robots find the shortest path in a grid.
- **Application**:
- Code a robot to navigate through a maze autonomously.

4/ **Computer Vision**

1. **Introduction to OpenCV**:
    1. Learn basic image processing: edge detection, color detection.
    2. Example: Detect and track a colored object.
2. **Machine Learning Integration**:
    1. Use pre-trained models for object recognition (e.g., YOLO or MobileNet).

5/ Advanced Control Systems

- **PID Control**:
- Theory: Proportional, Integral, and Derivative control.
- Practical Example: Use PID to stabilize a two-wheeled robot.
- **Adaptive and Predictive Control**:
- Basic introduction to advanced algorithms for dynamic environments.

6/ Collaborative Robots (Cobots)

1. **Human-Robot Interaction**:
   1. Detect obstacles dynamically and stop or re-plan paths.
   2. Case study: Collaborative robotic arm in industrial assembly.

7/ AI in Robotics

- **Deep Learning Basics**:
- Use convolutional neural networks for object recognition.
- Example: Train a robot to sort objects by type.
- **Reinforcement Learning**:
- Explore how robots learn by trial and error.