# IR Beacon Tracking Project Proposal

Leen Alshorafa (laa2202) & Kuan Zhang (kz2504)

**Overview**

The core of this project is a hardware accelerator for detecting and tracking an infrared (IR) beacon using a camera connected to an FPGA development board. The system is designed to locate the beacon in real time and estimate its position within the camera's field of view. Such a system could be used for applications such as guiding drones toward landing markers or tracking objects equipped with IR emitters.

The FPGA hardware accelerator will process the incoming camera image stream and identify the beacon using simple image-processing techniques. Because the beacon will appear as the brightest object in the image, thresholding and blob detection can isolate it efficiently. The accelerator will compute the centroid of the detected beacon and output its position within the image frame.

Software running on the ARM processor of the DE1-SoC board will read this centroid position and convert it into angular coordinates relative to the camera. These coordinates may then be transmitted to another device or used to control a tracking system.

---

**Frame Rate and Resolution**

The system will operate on a video stream with a target resolution of 640×480 pixels at approximately 30 frames per second. The accelerator will process pixels as they arrive from the camera rather than storing full frames in memory. This streaming approach minimizes memory usage and allows the design to operate entirely within the FPGA's on-chip resources.

Because the beacon detection algorithm relies on simple thresholding and connected-component analysis, the computation required per pixel is small enough to be performed in real time at the camera's pixel clock rate.

---

**Image Processing**

Beacon detection will consist of several stages:

First, the FPGA will perform thresholding on the incoming pixel stream to produce a binary image in which bright pixels correspond to potential beacon locations. The threshold value will be adjustable through a register written by the processor.

Next, runs of adjacent foreground pixels will be grouped together to form candidate regions. These regions will be combined across scan lines to form connected components representing possible blobs.

For each detected blob, the hardware will accumulate the number of pixels and the sum of their coordinates. At the end of each frame, the blob with the largest area will be selected as the beacon candidate. Its centroid will be calculated and made available to the processor.

---

**Hardware–Software Interface**

The FPGA accelerator will expose a small set of memory-mapped registers that contain the centroid coordinates of the detected beacon and a status flag indicating whether a valid blob was found. These registers will be accessed by software through the Lightweight HPS-to-FPGA bridge.

Each frame, the processor will read the centroid values and convert them into angular offsets relative to the camera's optical axis using a simple pinhole camera model. These angular values can then be transmitted over a serial interface or used for further control logic.

---

**Optional Debug Output**

To aid debugging, the system may optionally display the thresholded image on the VGA output of the DE1-SoC board. In this mode, the detected centroid will be overlaid with a crosshair so that the tracking behavior can be observed visually.

---

**Major Tasks**

• Determine camera interface method and validate video capture from the IR webcam
• Implement thresholding and blob detection pipeline in FPGA hardware
• Design hardware logic for centroid calculation and blob selection
• Develop software running on the ARM processor to read centroid data and compute angular coordinates

• Create a desktop prototype of the detection algorithm to validate performance before implementing hardware
• Build a simulation testbench for the FPGA modules
• Integrate the system and demonstrate real-time IR beacon tracking