Tag detection

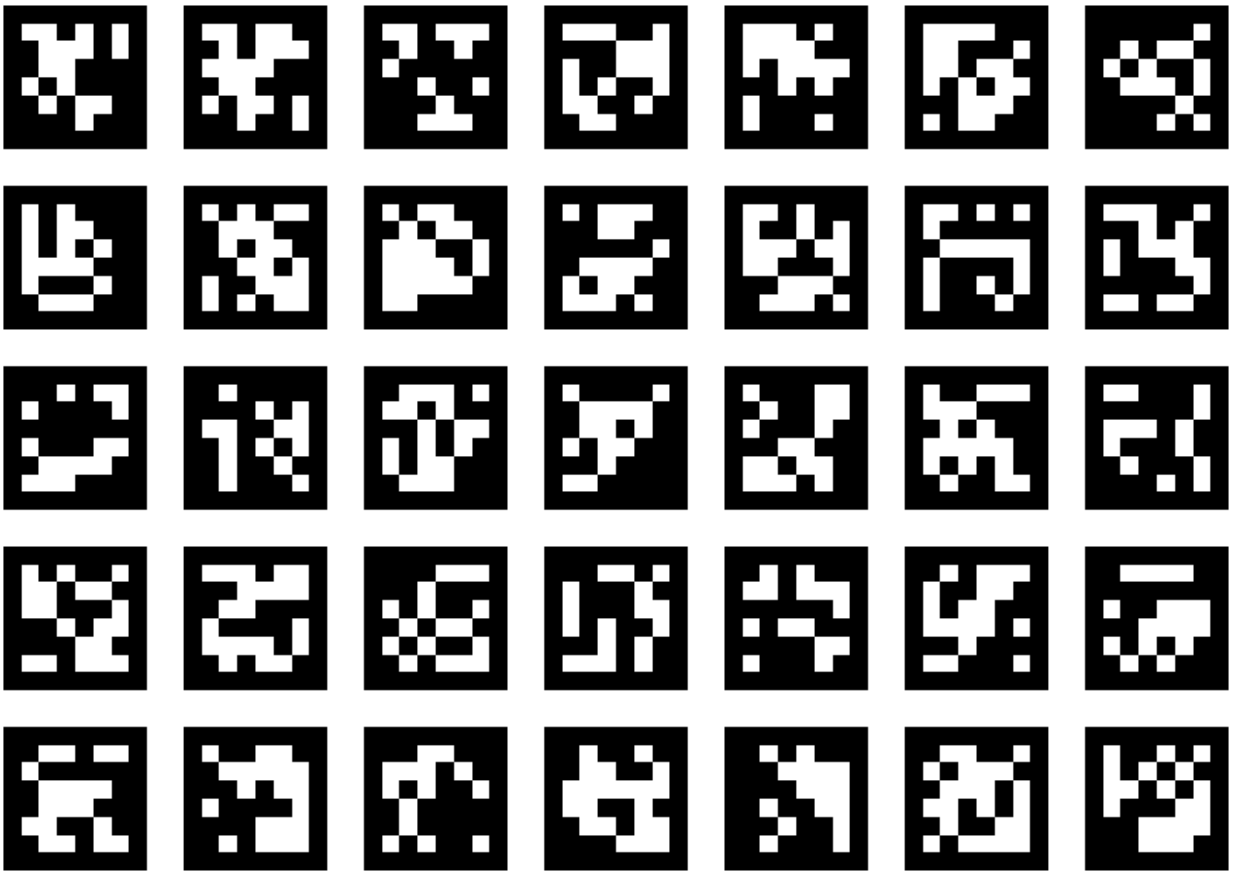
Develop a ROS package in c++ to detect 2D tags.

- Description
  1. Create a compilable ROS package, named as tag_detector. (10 points)
  2. Print the Apriltags attached, and record a 20 seconds video of the tags from different angles and distances with your smartphone. (10 points)
  3. In the first ROS node, load the video, and create a topic publisher (topic name "/image_raw") to publish the images in an order. (10 points)
  4. In the second ROS node, create a topic subscriber to "imag_raw" (10 points)
  5. In the second ROS node, implement a rectangle detector to detect the boundary of 2D square tags in the image, visualize the detected rectangle with OpenCV tool, and measure computation time. (20 points)
  6. In the first ROS node, add a Gaussian blur on the input image, and create a ROS service server "/set_blur_window_size" to adjust the Gaussian blur window size. (10 points)
  7. In the second ROS node, evaluate the sharpness of the image by using FFT. (10 points)
  8. In the second ROS node, if the image is detected highly blurred (fail to detect a rectangle), implement a deblur filter before applying rectangle detector. (10 points)
  9. Bonus: record another video of the tags in a dark room, with phone flashlight on, to create an uneven illumination. In the second ROS node, design an appropriate filter to achieve a successful detection.

- Skills
  1. C++ (Object oriented programming is preferred.)
  2. ROS
  3. OpenCV
  4. A good code structure and style is a must. (No absolute path to a file, so we can run your submission.)

- Final Submission:
  1. Create a github repo and upload your package to a repo.
  2. Name of the company "Bito" should not be there anywhere in your code or in repo.
  3. It is advised to do frequent commits during code development.
  4. Develop each feature in a separate branch and merge back to master after module test is a bonus.
  5. Final submission is a link to the repo.

- Tag mosaic



Files

tag-mosaic.png	4.52 KB	02/22/2019	Jin Dai