Calibrate camera intrinsics with circle pattern

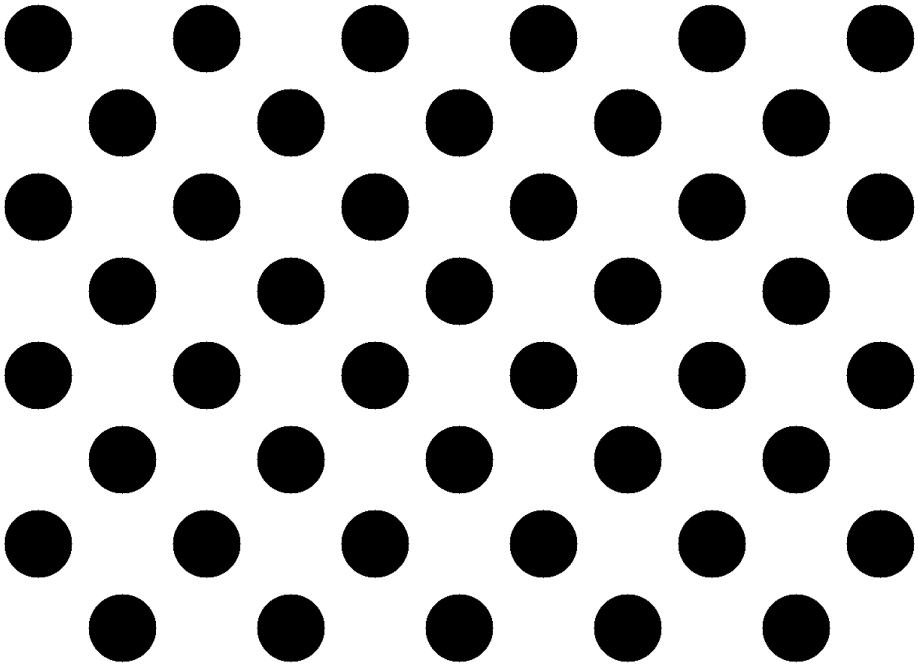
Write a ros node in c++ to calibrate camera intrinsics with circle pattern.

- Description
  1. Create a ros package (10 points)
  2. Print the circle pattern attached, and take 20 pictures of the pattern with your smartphone (10 points)
  3. In the first ros node, create a topic publisher (topic name "/image_raw") to publish these 20 pictures 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, detect the circle grid in an image. HINT: findCirclesGrid in opencv (10 points)
  6. In the second ros node, calibrate camera intrinsics. HINT: calibrateCamera in opencv (10 points)
  7. In the second ros node, write camera intrinsics calibration result to a standard yaml file. (10 points)
  8. In the second ros node, create a ros service (service name "/calibrate"). (10 points)
  9. In the second ros node, in "calibrate" service callback, start listening 20 pictures from "image_raw" topic, conduct a calibration and save to a yaml file. (20 points)
  10. Bonus: Subpixel refine for circle grid detection result. (10 points)
  11. Bonus: Implement Image pyramid for circle grid detection to speed up. (10 points)

- Skills
  1. C++
  2. ROS
  3. OpenCV
  4. A good code structure and style is a must.

- Code Hosting
  Create a github repo and upload your package to a repo. Final submission is a link to the repo.

- Circle pattern



Files

pattern_acircles.png	7.68 KB	01/23/2019	Jin Dai