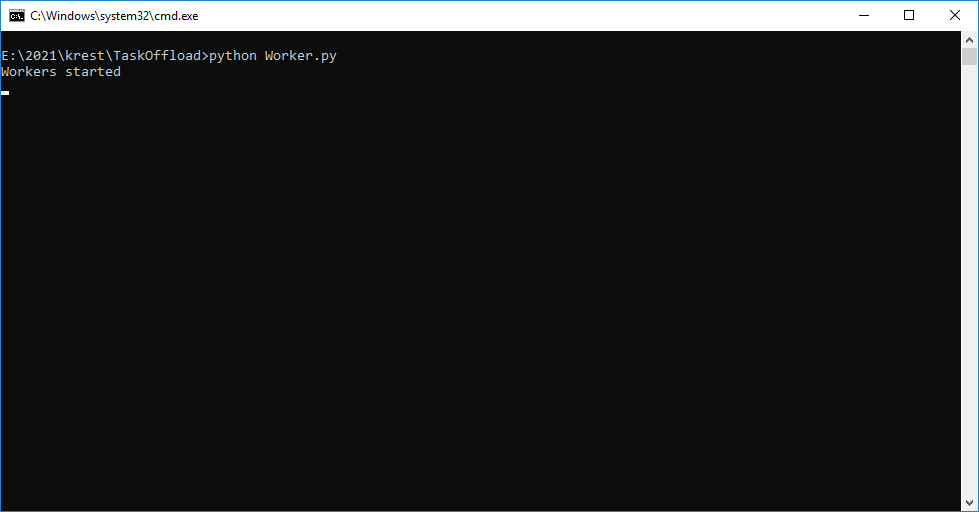
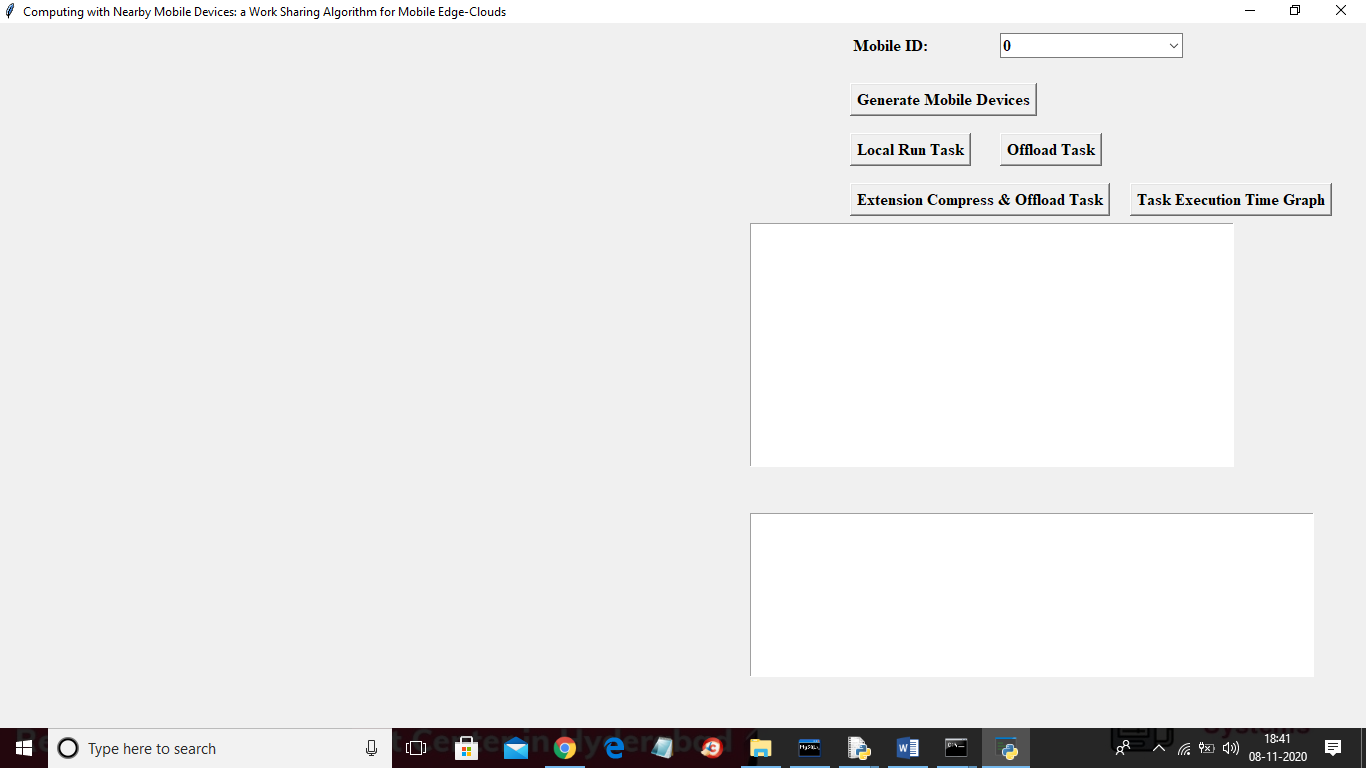
Computing with Nearby Mobile Devices: a Work Sharing Algorithm for Mobile Edge-Clouds

In this paper author is using nearby devices to offload and process task and in propose work author is not using any compression technique which can compress request data and then send to nearest devices for computation and if we compress request data then request size will be reduce and reached to nearby devices faster and nearby devices has to work on compress smaller data so processing time also be reduce. Below screen shots shows compression technique obtained output faster than propose work.

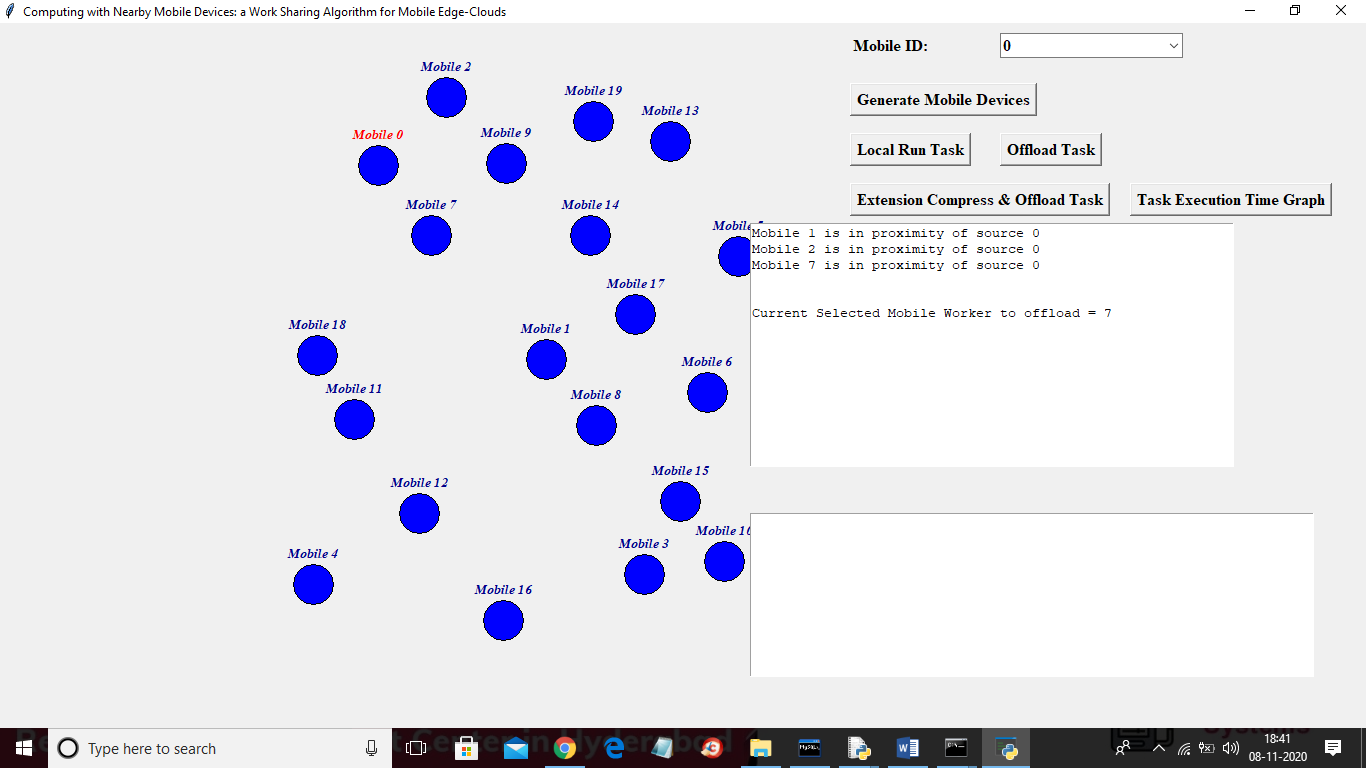
To run extension double click on ‘run\_worker.bat’ file to stat worker and to get below screen



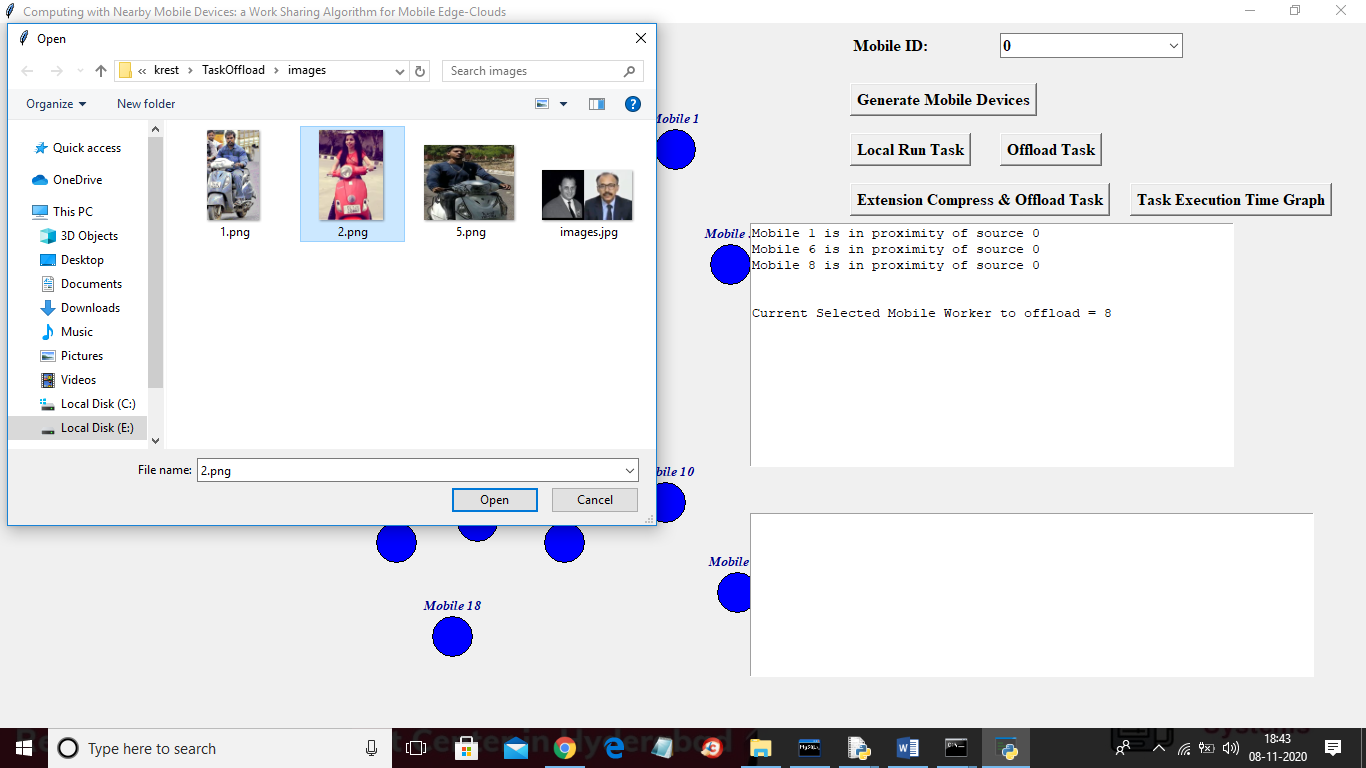
In above screen worker is started and now double click on ‘run.bat’ file to start mobile task offloading simulator



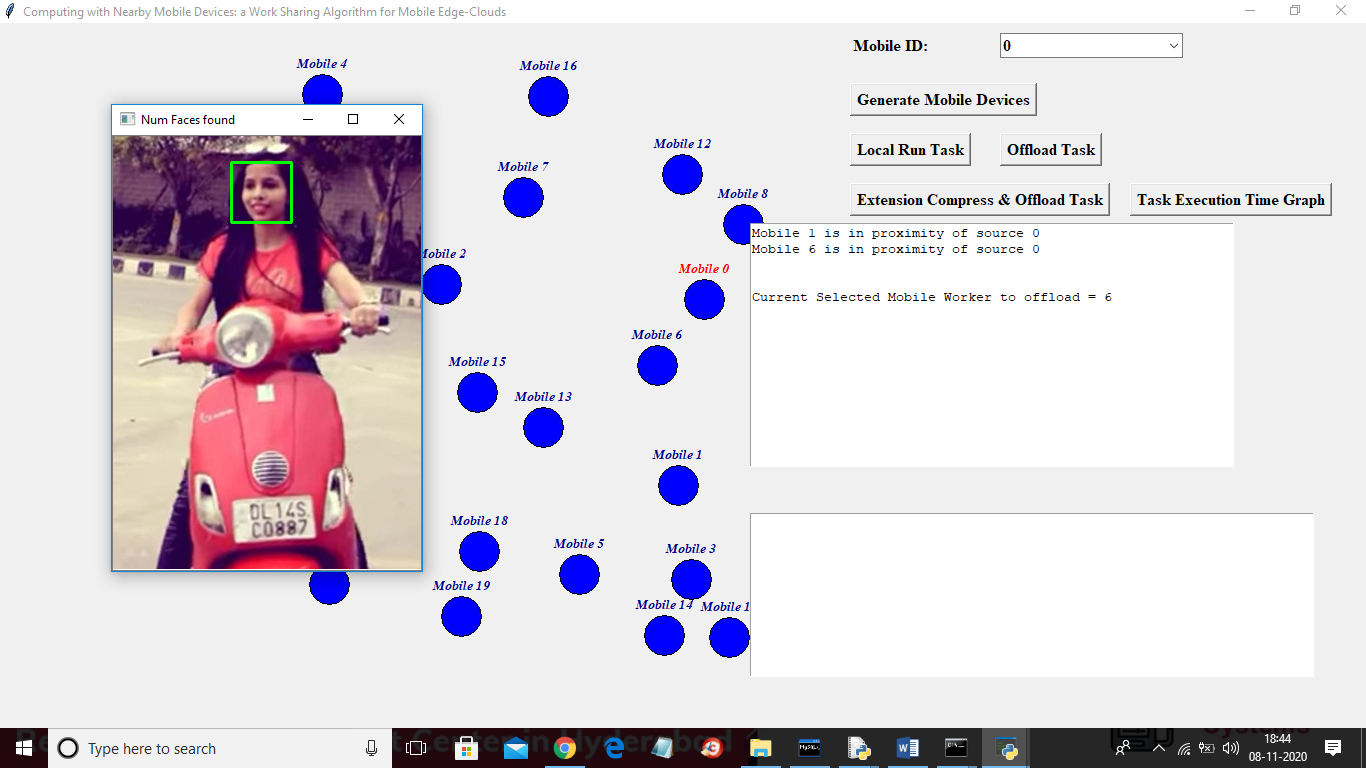
In above screen click on ‘Generate Mobile Devices’ button to generate mobile devices and to get below screen



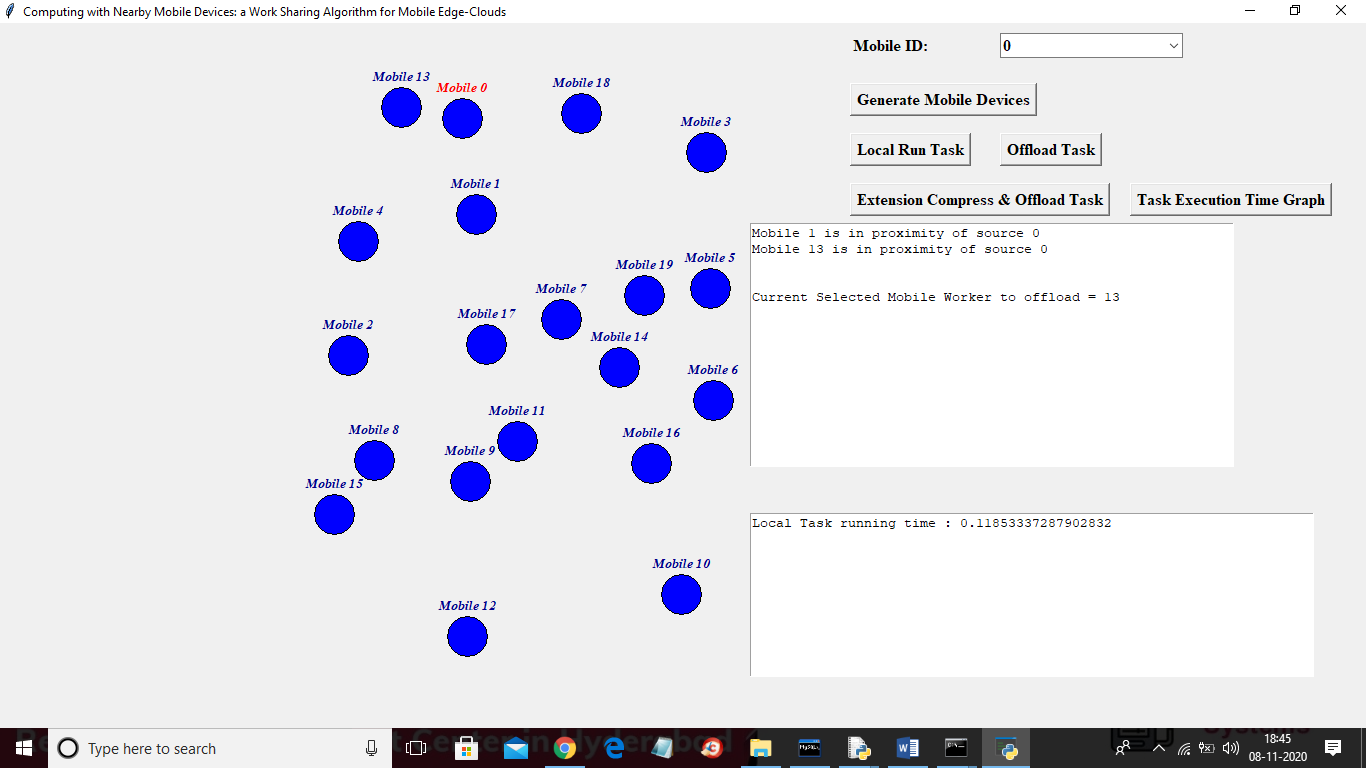
In above screen each circle will represent like a mobile and then select any mobile from Mobile ID drop down box and then click on ‘Local Run Task’ button to upload task to nearby devices and in this project we are uploading images and then offload to nearby devices which will detect faces and send back to source mobile



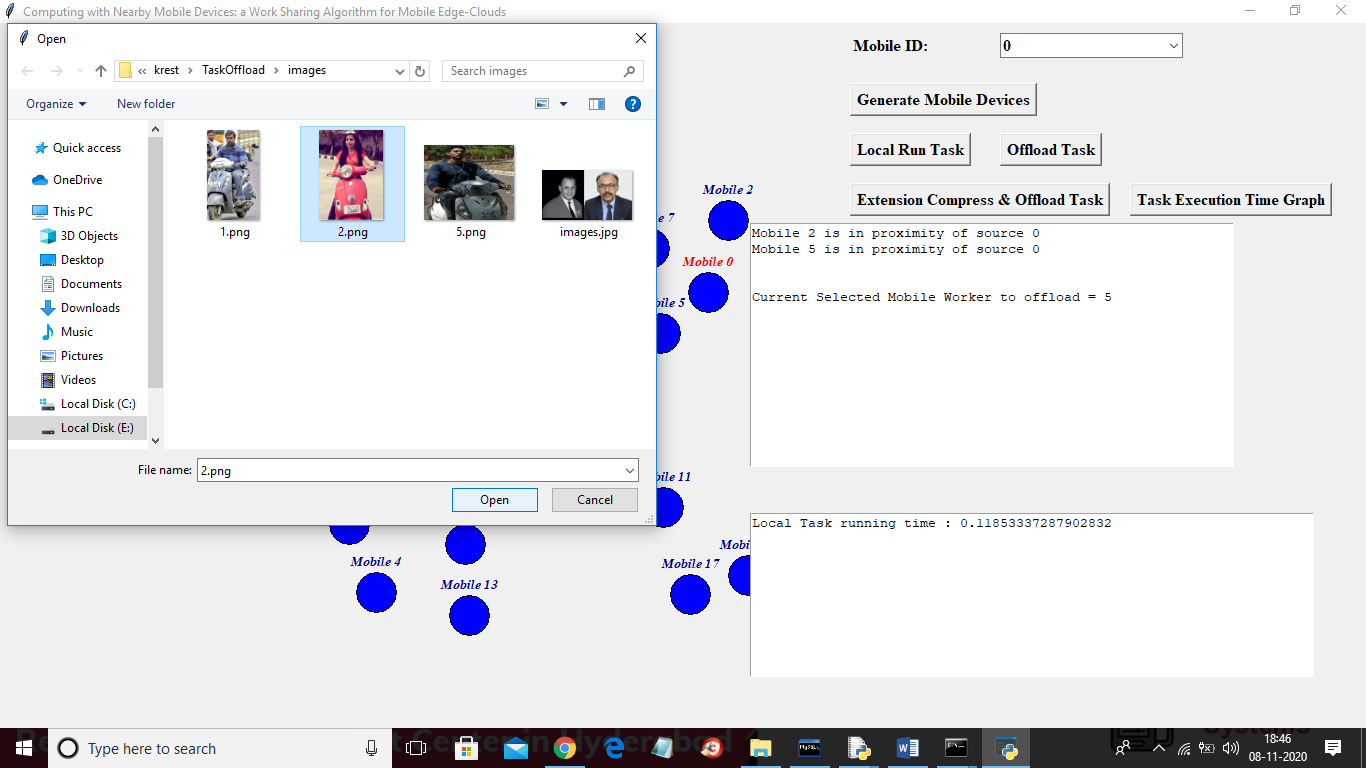
In above screen selecting and uploading ‘2.png’ and then click on ‘Open’ button to offload task and to get below result



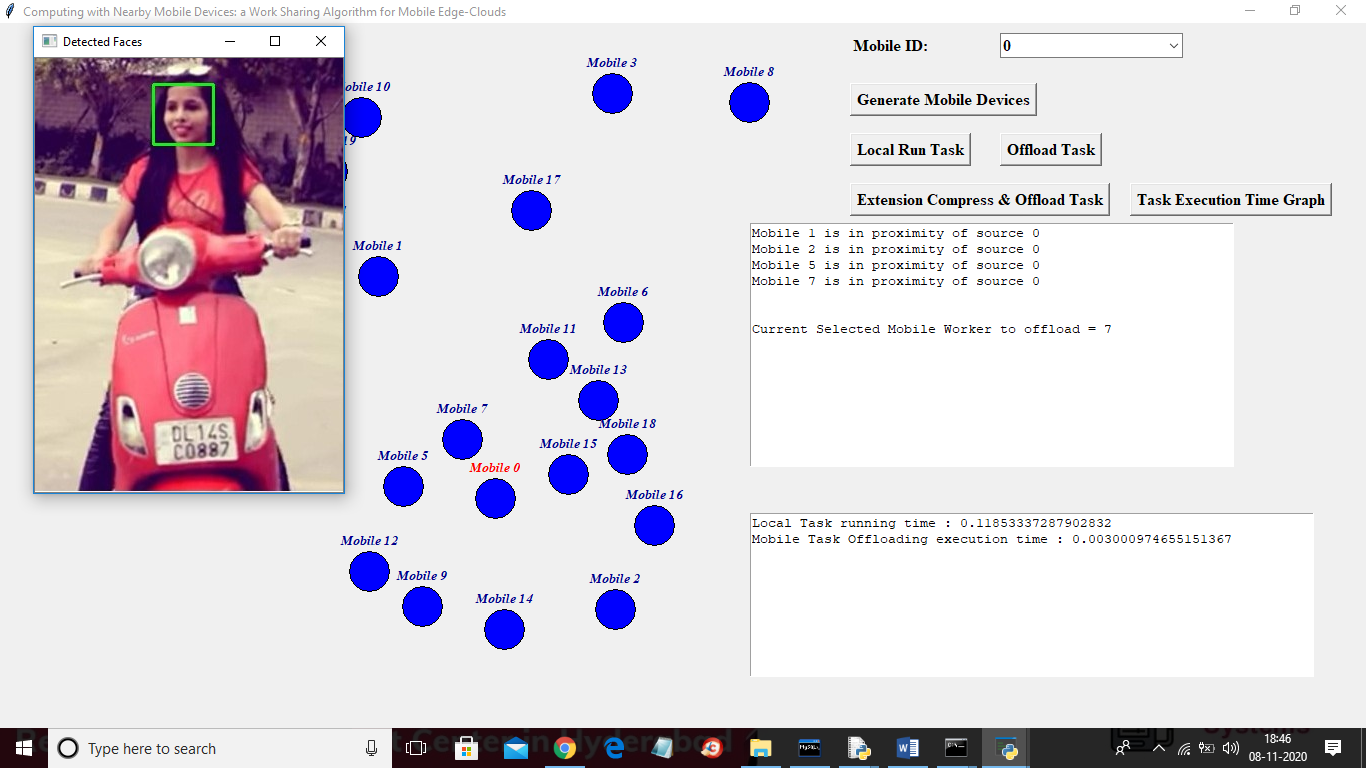
In above screen devices run locally to detect face from uploaded image and in below screen we can see total time taken by local run



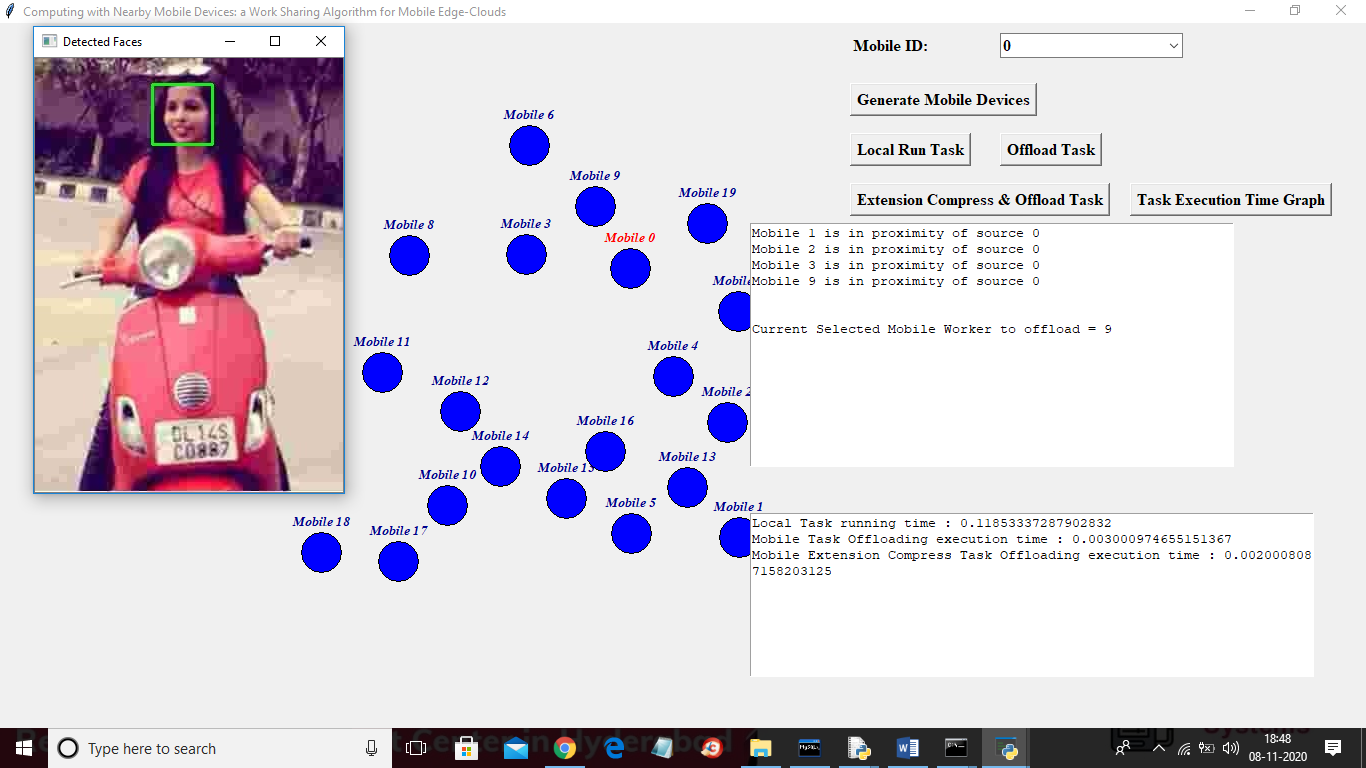
In second text area we can see total execution time taken by local run is 0.11 milli seconds and now click on ‘Offload Task’ and upload same image and offload to nearby devices



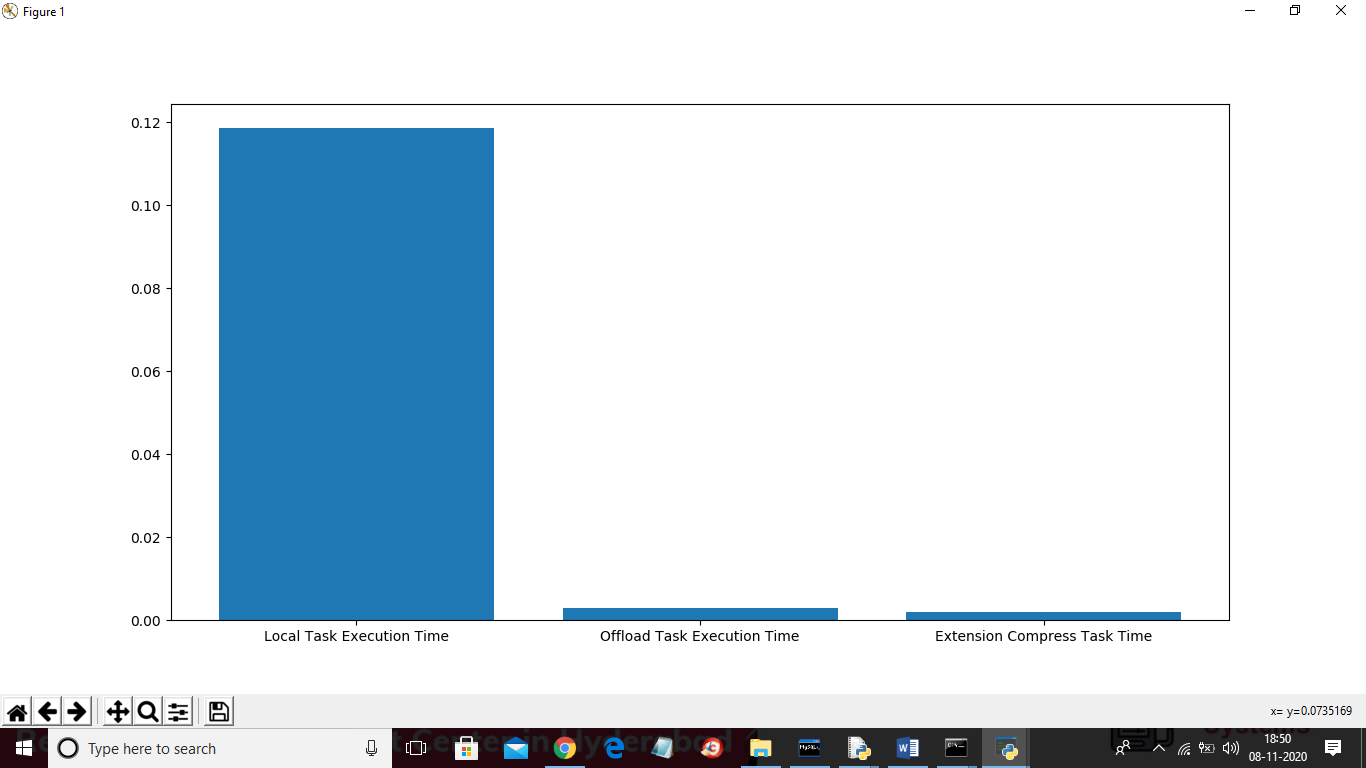
In above screen uploading same image and below is the result



In above screen source mobile offload task to nearby devices and in second text area we can see task offloader propose work took 0.003 Milli seconds and now click on ‘Extension Compress & Offload Task’ button to upload compress task and get below result



In above screen for extension compress task also we got same result and in second area we can see compress task offloading took 0.002 milli seconds which is faster than local and propose normal task offloading and now click on ‘Task Execution Time Graph’ button to get below graph



In above graph x-axis represents technique name and y-axis represents execution time in milli seconds and in all techniques extension compress task offloading took less execution time