# Nucleus - A P2P Image Sharing App Employing Compression

## CS4089 Project

## End Semester Evaluation

Anand M.P, Ronn George Jacob, Sreeraag Mohan T
Guided By: Ms. Pournami P.N

November 16, 2015

# Outline

# Introduction

- Nucleus:
  - an application that enables users to share images over an adhoc Wi-Fi network
- Compressed images:
  - a lossy compression algorithm based on human visual perception measurements would be chosen with the help of a study comparing existing lossy compression algorithms
- Groups:
  - users would be able to create groups allowing them to interact with other users of Nucleus for transfer of compressed images.

# Problem Statement

- To implement a peer-to-peer image sharing application that uses an optimal compression algorithm to send compressed image files over an ad-hoc Wi-Fi network.

# Literature Survey

- The concepts of Android were studied from Neil Smyth's Android Studio Development Essentials [1] and Google's official developer site and public forums [2].
- Wi-Fi direct and the Wi-Fi P2P functionalities were explored using documentation provided here [3].
- JPEG image compression standards were provided by Wallace [4].
- JPEG2000 and its comparative benefits were learnt from Singh and Srivastava's paper [5].
- Working and advantages of SSIM were studied from Wang, Bovik et.al's paper [6].
- Power Optimization Techniques : Yereaztian and Luthiger [7], Udacity's online course [8]
- UML and SRS documentation: Lethbridge and Laganiere's textbook [9].

# Work Done I

- Identifying Compression Algorithms
  - Considered three compression algorithms: JPEG, JPEG2000, and WebP.
  - JPEG2000, due to its inherently complex code and large storage footprint, was not considered for the comparative study.
- Comparative study of lossy compression algorithms using SSIM metric
  - **Aim:** To choose a lossy compression algorithm that suffices the requirements
  - Done with three types of images : greyscale, RGB landscape and portrait images

# Work Done II

- SSIM calculated with reference to original, uncompressed image at various bitrates

$$\mathrm{SSIM}(x,y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}$$

- **Result:** JPEG places minimal strain on the limited processing capabilities and is the best choice for Nucleus

- Software Requirements Specification
  - Developed the use case diagrams and modelled the various interactions that a user would have
- Power Optimization
  - Optimization techniques and performance tradeoffs were evaluated
  - Introduced themes: light and dark themes
  - Minimize number of threads to reduce impact on CPU usage

# Future Work

- Enter implementation stage as soon as mockups are designed
- We aim to use a VCS such as *GitHub*
- More battery optimization techniques will be explored during implementation, for eg., *Battery Historian*

# References I


[1] Neil Smyth, *Android Studio Development Essentials*, Second Edition

[2] Android Developers, developer.android.com, Google Inc., 2015

[3] Android Wi-Fi P2P, http://developer.android.com/reference/android/net/wifi/p2p/pacl summary.html

[4] Wallace, G.K., *The JPEG still picture compression standard* , in Consumer Electronics, IEEE Transactions on,vol.38, no.1, pp.xviii-xxxiv, Feb 1992

[5] Singh, R.; Srivastava, V.K., *JPEG2000: A review and its performance comparison with JPEG* , in Power, Control and Embedded Systems (ICPCES), 2012 2nd International Conference on, vol.,no., pp.1-7, 17-19 Dec. 2012

# References II


[6] Zhou Wang; Bovik, A.C.; Sheikh, H.R.; Simoncelli, E.P.,*Image quality assessment: from error visibility to structural similarity* , in Image Processing, IEEE Transactions on, vol.13, no.4, pp.600-612, April 2004.

[7] Chris Yereaztian, Jurg Luthiger, *Android Best Practices to Improve Battery Efficiency*

[8] Colt McAnlis, Chris Lei, *Android Performance - Optimizing Apps for Speed and Usability,* udacity.com/course/android-performance-ud825

[9] Lethbridge, Timothy and Laganiere, Robert, *Object-oriented software engineering: Practical Software Engineering Using UML and Java,* Mcgraw-Hill College, 2001