CS M117 Winter 2018
Special Wireless Experiment
Share Album

Team CS8Claps

Jeffery Yang
704781854
jefferyy@ucla.edu

Junu Park
205041631
junupark96@gmail.com

Scott Lee
204955724
bwlee6736@gmail.com

Abstract

Share Album is an Android application that share a photo in one-to-many network. It was made in Android Studio. The core technology behind the application is Wi-Fi Alliance's Wi-Fi Direct. Wi-Fi Direct, initially called Wi-Fi P2P, is a Wi-Fi standard that enables devices to connect easily with each other without requiring a wireless access point, such as a router. Hence, Wi-Fi Direct is single radio hop communication. It is useful for everything from internet browsing to file transfer, and to communicate with one or more devices simultaneously at typical Wi-Fi speeds. One advantage of Wi-Fi Direct is the ability to connect devices even if they are from different manufacturers. Only one of the Wi-Fi devices needs to be compliant with Wi-Fi Direct to establish a peer-to-peer connection that transfers data directly between them with greatly reduced setup.

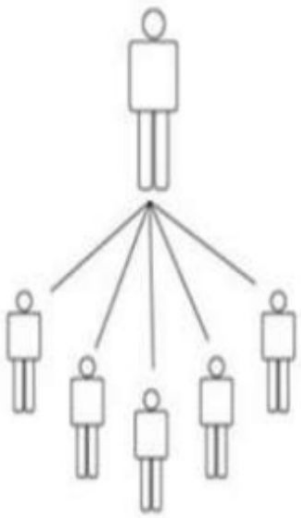


Figure 01: One-to-Many network [1]

Motivation

Imagine you see an exotic landscape while traveling with your friends. You take a picture of it, and the other friends are asking for it. It is such an annoying job to send the photo to all your friends. What if the number of mates are like 10 people? We wanted to make a smart solution for this situation. With Share Album, sharing a photo is instant.

There are many applications that conduct data transmission. However, most of them require internet connection, such as wifi or a hotspot. We wanted to transfer data across devices offline and locally. Therefore, this application, Share Album, can be useful in a place where cellular-internet and wifi don't exist. In addition, the default Wi-Fi Peer-to-Peer API only supports one-to-one network, which is not ideal for data transmission. Therefore, we extended into one-to-many network. It allows that a group owner can spread a photo into unlimited devices.

Furthermore, the reason why we chose Wi-Fi Direct over Bluetooth is the data rate of transmission. While Bluetooth's data rate is up to 25 Mbps, Wi-Fi Direct allows the data rate of up to 250 Mbps. Beside the advantage of data rate, another advantage of Wi-Fi Direct is the greater range of connection. Wi-Fi Direct devices work with ranges up to 200 meters; it is three times farther than Bluetooth.

Functionality

The overall job of Share Album is for a group owner to share a photo to the all group members. One core functionality is to make a group of peer connections. With a group owner, a number of peers can be connected. Connection can be done by Discover and Connect methods.

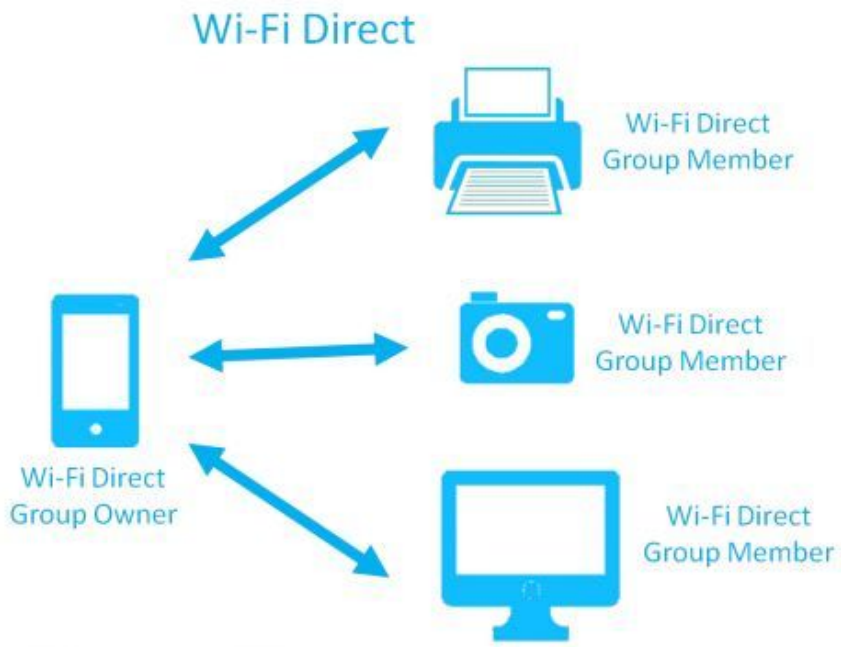
In addition, the list of connection in the group can be shown using “See Group” button. In the group owner device, it will show the list of peers connected to the owner. Another core functionality is to share a photo. The group owner spread a photo into the all peers connected to the owner. The photo can be a new photo or an existing photo stored in Gallery. The photo sent by the group owner will be popped up on the clients’ screen and stored in the clients’ Gallery under ShareAlbum folder. Beside the core functionalities, there are some minor but necessary functionalities: Enable/Disable Wifi, See group, Disconnect, and Remove group.

Enable/Disable Wifi method allows users to directly jump to the Setting to easily turn on or off the Wifi. See group is for seeing who are in the group. It displays the device names that are in the group; therefore, the group owner can manage the group easily. Disconnect is for clients to leave the group that they are currently in. Remove group is for a group owner to destroy the group it initiated; it is useful when photo sharing is complete.

Wireless Technology

The wireless technology we used in our project is Wifi Direct. Wifi Direct is also sometimes known as Wifi P2P. Wifi Direct is a standard which expands upon the traditional Wifi standard. When using traditional Wifi, one must connect to a designated router in order to access the internet. With Wifi Direct, two devices can establish Wifi connection without a router and transfer information. Wifi Direct allows this type of peer-to-peer communication by embedding a software access point into devices such as mobile phones and computers. For example, a smartphone would be able to serve as a software access point, and devices within range of the smartphone can connect to it directly. When establishing the connection, Wifi Direct uses Wifi

Protected Setup (WPS). A secure way of setting up the connection between two devices is using the push button method, in which one devices sends a connection request and the other device accepts the connection. The push button method eliminates the need to enter a password to connect.



Wifi Direct uses single radio hop communication, unlike wireless multihop ad hoc networks. This results in one-to-many data transfer where there is one group owner in the network and the rest of the devices are clients. The group owner can communicate with all the clients, and all the clients can communicate with the owner. Wifi Direct is being into integrated into a variety devices, such as phones, laptops, printers, game consoles, and televisions.

There are many different applications of Wifi Direct. A big application, and one we used for our project, is file transfer. Transferring files using Wifi Direct allows for quick transfer of information without using cables or traditional Wifi. For example, instead connecting a computer to a printer to print a file, the computer can connect to it using Wifi Direct and print the file.

Another application for Wifi Direct technology is peripheral communication, such as wireless remote controls, displays, speakers. etc. For example, the Roku is able to be remote controlled using Wifi Direct. Wifi Direct can also be used for screen sharing, internet sharing, and smart home applications.

Implementation

We implemented our app for Android devices using Android Studios and our project was written in Java. We used Android Wi-Fi Peer-to-Peer (P2P) API, which is Android's implementation of Wi-Fi Direct. The WifiP2pManager class allowed us to interact with our devices' Wi-Fi hardware to perform actions such as discover peers, connect to peers, get the current connection information, create a P2P group, get the current P2P group information, and remove or leave the current P2P group, all of which we implemented into our app.

One of the most important functionalities from the ones provided to us by the Android Wi-Fi P2P API was the P2P group functionality, since we desired a one-to-many connection network instead of one-to-one. In a P2P group, the group owner was the obvious best candidate for our dedicated photo taker, since it is the only device directly connected to all of the group members. It is also the only device that is able to see who was in the current P2P group. However, this API was meant to be used in a certain way which conflicted with our plans. Apparently, the group owner was meant to act as a sort of server, and the group members were meant to be clients that could freely connect to the server.

More specifically, the API allowed the group owner to obtain some information about the group members such as their device names and MAC addresses but not their IP addresses. The

only IP address that could be easily obtained was the group owner's by any device in the group. Hence, initially there was no way for the group owner, our designated photo taker, to be able to freely connect and send data to the group members.

Our solution to this was to have each group member send the group owner an empty message when they first connect. Then the group owner would obtain the member's IP address from the socket and store it in a set for later use.

To implement data transfer, we utilized sockets with TCP/IP for reliable and ordered data streams since we are handling file transfer. We have every device start a server socket upon creating or joining a P2P group. This is done for the group owner so that it can receive the members' IP addresses and done for the group members so that they can receive image files from the owner. Once all the connections have been established, we are ready to share photos! Every time the owner selects a photo from the gallery or takes a new one with the camera, it loops through the stored members' IP addresses and does the following for each address:

1. Opens a socket with the current IP address
2. Opens an input and output stream to the socket
3. Writes the image file data to the stream while the receiver reads from it
4. Closes the streams and then the socket

If the owner took a new photo with the camera, we also save it to its gallery.

On the receiving end, we have the group member accept the socket and read the image file from the stream. Once the data transfer is complete, we save the image to a custom directory within the device's stock gallery app and then display it to the screen.

Demo

https://www.youtube.com/watch?v=BUrWsPbABEE

Contribution

All three members evenly contributed the amount of work into Share Album project. We worked together on the brainstorming of the project, choosing IDE, design, research, implementation, and testing. More specifically, the following shows who focused more on what sections; however, it does not mean that a group member did not do other works that are not listed:

Jeffery Yang focused on research to obtain basic idea and sample codes of API we needed. In addition, he kept thinking of new features that would improve our application. For example, although we decided not to add at the end, he suggested to add a dedicated cloud server and set up the server. The application uploads the photo up to the server when the group owner sends out a photo to the group members. The group owner can download it anytime later.

Junu Park focused on the implementation of the project. This includes implementation of the Android Wi-Fi Peer-to-Peer functionalities and file transfer. He worked on adding the ability to discover peers, create a P2P group, connect to peers, see the current group information, and remove/leave the current group to the app. Together with Scott, he implemented the data transfer functionality with use of sockets and accomplished successful data transfer from the group owner to all of its members.

Scott Lee focused on the user interface, group planning and organizing, and debugging. He designed the overall layouts of the application. For example, the initial version of the

application had multiple activities; however, it ended up with one activities with data fragments and action bar items. In the process of modifying the layouts, he integrated the initial implementation into the fragments successfully. Therefore, the user interface/experience significantly increased. In addition, by dividing works into group members properly and setting deadlines, the project was done efficiently. He also tested possible situations to find exceptions, and then fixed bugs.

How-to-use

The following procedure is shown in the demo video. You can refer to it.

i. Import the project into Android Studio.

ii. Install the application on devices that you want to test with.

    i. In order to test the code, you might need at least three android phones.

iii. In action tap, click “Create Group” in a device that you want to set as a group owner.

iv. After successfully creating group, in action bar, click “Discover” for all devices that you want to connect with including the group owner.

v. In the group owner device, find the client devices in the list of discovered devices and click “Connect.” Then, an invitation will pop up in the selected client device. Click “Accept.” Repeat for all client devices.

vi. After successfully connecting devices, in the group owner device, send a photo.

    i. Option 1: New photo

    When you click “New photo”, you can take a picture using the camera. After taking it and clicking “Save”, it will be sent to all clients.

    ii. Option 2: From Gallery

    When you click “From gallery”, you can select an existing picture in the Gallery. Click a photo that you want to share, and then it will be sent to all clients.

Works Cited

Wyckoff, Andrew, et al. Digitalisation and the Future of Work (Part 1). Oct. 13, 2016. Mar. 13, 2018.

Wi-Fi Alliance. Wi-Fi Direct. Wi-Fi Alliance. Wi-fi.org. Mar. 13, 2018 https://www.wi-fi.org/discover-wi-fi/wi-fi-direct