

# Poster: Camera Images Offloading in Low-resource Wireless Networks

Wei Xiong, Mariya Zheleva  
University at Albany, SUNY  
{wxiong, mzheleva}@albany.edu

## 1. MOTIVATION AND METHODOLOGY

Camera snapshot images are widely used in IoT applications such as smart farming and weather monitoring, which generate large volumes of image data. However, when these settings are in rural areas or poorly-performing networks, the images offloading usually exhausts the limited network resources. While we can certainly revamp the network links, we can also optimize the payload at the same time[1]. From our site survey we see that for consecutive camera snapshots, there is a notable similarity among the contents, especially from fixed-positioned cameras used in IoT environments.

**Methodology.** We design a system that computes and transfers only the “difference” between consecutive images, thus drastically reducing the network offered load. Original images are then restored at the server side. Our system uses a middlebox to intercept camera images, compute the “difference” across image series before uploading them to the server. The middlebox is plug-and-play and can be deployed on a cheap single board computer near the cameras; for example, on an OpenWRT-based access point serving the cameras. Also, we use another counterpart module on the server to “decode” the payload and restore the original image. In this early implementation, we use *Pillow* as the imaging tool and *Scapy* to manipulate packets passed through. Because of lossy compression operations used in both “encode” and “decode” stages, the restored image may introduce some small error compared to the original image, but the approach significantly reduces the payload transferred which is a good choice to alleviate poorly performing networks.

## 2. PRELIMINARY EVALUATION

We choose 330 traffic cameras in New York[2]. For each camera, we download 100 consecutive images at a 10-second interval. Our selected cameras provide a wide diversity of images, ranging from streets and highways to bridges and tunnels. From any two consecutive images of the same camera, we subtract latter image  $B$  from previous image  $A$  to

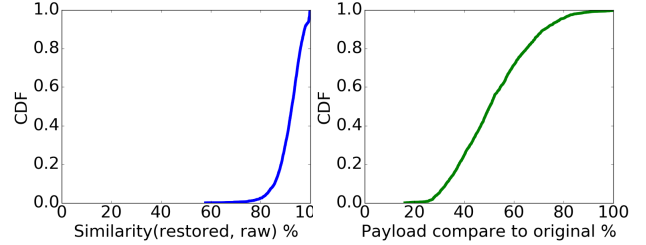


Figure 1: Accuracy

Figure 2: Reduced payload

get the difference (aka, the payload in our proposed system). We calculate the size of reduced image  $\delta$  comparing to  $B$  as shown in Fig.2. We then add  $\delta$  and  $A$  to get the restored image  $B'$ . We calculate the **structural similarity** between  $B$  and  $B'$ , denoted as accuracy in Fig.1.

Note that there is a difference due to what we called one-step optimization and multi-step optimization. Denote  $A$ ,  $B$  and  $C$  as consecutive images,  $f_1, f_2$  as decoding and encoding respectively. The above result shows

$$Sim(C, f_2(B + f_1(C - B))) \quad (1)$$

But instead of using raw image  $B$ , we do

$$Sim(C, f_2(B' + f_1(C - B'))) \quad (2)$$

where  $B' = f_2(A + f_1(B - A))$

Apparently (2) does not necessarily yield same accuracy as (1) does. However (2) leads to a higher sum of reduced payload. Therefore in order to achieve the result shown, we assume using one-step optimization, meaning the middlebox should offload the raw image in every other cycle. While it can certainly perform multi-steps optimization, the trade-off between the compound error on similarity and the sum of reduced payload could be one aspect of our further study. Also, we notice the worst accuracy comes from camera images with random striping noise, which should be further studied in order to deal with noise and other external factors.

## 3. REFERENCES

- [1] W. Hu et al. The case for offload shaping. In *Proceedings of the 16th International Workshop on Mobile Computing Systems and Applications*, pages 51–56. ACM, 2015.
- [2] NYC Traffic Cameras. <http://dotsignals.org>.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

MobiSys'17 June 19-23, 2017, Niagara Falls, NY, USA

© 2017 Copyright held by the owner/author(s).

ACM ISBN 978-1-4503-4928-4/17/06.

DOI: <http://dx.doi.org/10.1145/3081333.3089316>