Technique for mobile image processing in the cloud cuts bandwidth use by more than 98 percent

November 13 2015, by Larry Hardesty

As smartphones become people's primary computers and their primary cameras, there is growing demand for mobile versions of image-

processing applications.

Image processing, however, can be computationally intensive and could quickly drain a cellphone's battery. Some mobile applications try to solve this problem by sending image files to a central server, which processes the images and sends them back. But with large images, this introduces significant delays and could incur costs for increased data usage.

At the Siggraph Asia conference last week, researchers from MIT, Stanford University, and Adobe Systems presented a system that, in experiments, reduced the bandwidth consumed by server-based image processing by as much as 98.5 percent, and the power consumption by as much as 85 percent.

The system sends the server a highly compressed version of an image, and the server sends back an even smaller file, which contains simple instructions for modifying the original image.

Michaël Gharbi, a graduate student in electrical engineering and computer science at MIT and first author on the Siggraph paper, says that the technique could become more useful as image-processing algorithms become more sophisticated.

"We see more and more new algorithms that leverage large databases to take a decision on the pixel," Gharbi says. "These kinds of algorithm don't do a very complex transform if you go to a local scale on the image, but they still require a lot of computation and access to the data. So that's the kind of operation you would need to do on the cloud."

One example, Gharbi says, is recent work at MIT that transfers the visual styles of famous portrait photographers to cellphone snapshots. Other researchers, he says, have experimented with algorithms for changing the apparent time of day at which photos were taken.

Joining Gharbi on the new paper are his thesis advisor, Frédo Durand, a professor of computer science and engineering; YiChang Shih, who received his PhD in electrical engineering and computer science from MIT in March; Gaurav Chaurasia, a former postdoc in Durand's group who's now at Disney Research; Jonathan Ragan-Kelley, who has been a postdoc at Stanford since graduating from MIT in 2014; and Sylvain Paris, who was a postdoc with Durand before joining Adobe.

Bring the noise

The researchers' system works with any alteration to the style of an image, like the types of "filters" popular on Instagram. It's less effective with edits that change the image content—deleting a figure and then filling in the background, for instance.

To save bandwidth while uploading a file, the researchers' system simply sends it as a very low-quality JPEG, the most common file format for digital images. All the cleverness is in the way the server processes the image.

The transmitted JPEG has a much lower resolution than the source image, which could lead to problems. A single reddish pixel in the JPEG, for instance, could stand in for a patch of pixels that in fact depict a subtle texture of red and purple bands. So the first thing the system does is introduce some high-frequency noise into the image, which effectively increases its resolution.

That extra resolution is basically meaningless—just some small, random, local variation of the pixel color in the compressed file. But it prevents the system from relying too heavily on color consistency in particular regions of the image when determining how to characterize its image transformations.

Patch work

Next, the system performs the desired manipulation of the image—heightening contrast, shifting the color spectrum, sharpening edges, or the like.

Then the system breaks the image into chunks—of, say, 64 by 64 pixels. For each chunk, it uses a machine-learning algorithm to characterize the effects of the manipulation according to a few basic parameters, most of which concern variations in the luminance, or brightness, of the pixels in the patch. The researchers' best results came when they used about 25 parameters. So for each 64-by-64 pixel patch of the uploaded image, each pixel of which could have one of three values, the server sends back just 25 numbers.

The phone then performs the modifications described by those 25 numbers on its local, high-resolution copy of the image. To the naked eye, the results are virtually indistinguishable from direct manipulation of the high-resolution image. The bandwidth consumption, however, is only 1 to 2 percent of what it would have been.

Applying the modifications to the original image does require some extra computation on the phone, but that consumes neither as much time nor as much energy as uploading and downloading high-resolution files would. In the researchers' experiments, the energy savings were generally between 50 and 85 percent, and the time savings between 50 and 70 percent.

"There are a lot of things that we're coming up with at Adobe Research that take a long time to run on the phone," says Geoffrey Oxholm, a research and innovation engineer at Adobe who was not involved in the project. "Or it's a big hassle to optimize them for every single mobile platform, so it's attractive to be able to optimize them really well on the

server. It's almost like cheating: You get to use a big huge server and not pay as much to use it."

"On the stuff that we've tried it on, it's working great," Oxholm adds. "The fact that they look good enough is troublesome to me. I don't really understand how this is possible. I think there's probably some lesson buried deep in here about what realistic images look like."

More information: Transform Recipes for Efficient Cloud Photo Enhancement: s3.amazonaws.com/mgharbi-asset … s_for.pdf?1446446728

This story is republished courtesy of MIT News (web.mit.edu/newsoffice/), a popular site that covers news about MIT research, innovation and teaching.

Provided by Massachusetts Institute of Technology