

# QuiltView: a Crowd-Sourced Video Response System

Zhuo Chen, Wenlu Hu, Kiryong Ha, Jan Harkes, Benjamin Gilbert,  
Jason Hong, Asim Smailagic, Dan Siewiorek, Mahadev Satyanarayanan

School of Computer Science, Carnegie Mellon University

## 1. INTRODUCTION

We demonstrate a crowd-sourced system called *QuiltView* that leverages the ability of wearable devices such as Google Glass to provide *near-effortless capture of first-person viewpoint video*. Recording a video clip merely involves touching the shank of your Glass device. The extreme simplicity of video capture can be used to create a new kind of near-real-time social network. In this network, users can pose brief queries to other users in a specific geographic area and receive prompt video responses. The richness of video content provides much detail and context to the person posing the query, while consuming little attention from those who respond. The QuiltView architecture incorporates result caching, geolocation and query similarity detection to shield users from being overwhelmed by a flood of queries.

## 2. SYSTEM ARCHITECTURE

QuiltView is a cloud-based service that is built using off-the-shelf Internet technology. Figure 1 shows the QuiltView architecture. At the heart of this architecture is a global catalog of users and queries, implemented with a SQL database. It includes details of all the queries that have been posed, and all the responses that have been received, including the YouTube link of the response. Both the uploading and viewing of videos are done using standard YouTube mechanisms that are wrapped inside QuiltView query and response software. The QuiltView catalog also includes details of users and their preferences regarding their willingness to respond to queries. A user can change these preferences at any time, with immediate effect.

The QuiltView catalog is used as a result cache that short-circuits query processing. If a query is deemed to be “close” to a recently-answered one and “recent enough,” the cached results for the earlier query are returned. The user posing the query is first shown the earlier query, meta-data about the query, and the number of available responses. He has the choice of accepting the cached results or insisting that QuiltView obtain fresh results for his query. Result caching greatly improves the speed of responses from the viewpoint of the user posing the query. It also reduces the burden of repeated queries on Glass users.

Queries are posed using a web interface to Google Maps. A user can zoom into a geographic region and then poses a query such as “Have you seen my dog?” along with a thumbnail image of his pet. The zoomed-in region on the map implicitly defines the scope of the query. Within this geolocated scope, a subset of the Glass users who have opted-in to receive QuiltView queries on a relevant topic (such as

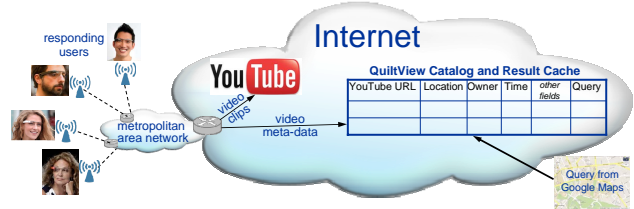


Figure 1: QuiltView Architecture

“pets” or “community help”) receive the query. When a user receives a query, he can decide to respond with a brief video clip of an appropriate part of the scene around him. Each video clip is uploaded into YouTube, and its link is displayed in the list of responses to the query. Meta-data about this video clip, matched to the original query, is entered into the global QuiltView catalog.

## 3. DEMONSTRATION DETAILS

Our demo will show a complete prototype of the QuiltView architecture. We will ask one to three participants to wear Google Glass and serve as video responders. Another participant will serve as the requester. Through our web interface, the requester can pose a query at his interest, such as “How exciting is HotMobile?”. He will also specify a region through Google Maps so that QuiltView users within that area will be queried. If the responders are within that area, the query will pop up in their Glasses within a few seconds. With only one touch of the Glass, a responder will record a ten-second video which will then be shown to the requester.

The participants are also able to set their preference settings, such as specific topics, the acceptable volume of queries during some period of time (e.g., queries per day or queries per hour). The QuiltView system will respect user preferences rigorously and within that constraint spread the burden randomly across users. We will demonstrate how preference settings can be done and how the system does load balancing among responders. We will also demonstrate the functionality of query similarity detection. If a new query proposed by the requester is similar to an old query in our database, the video responses to the old query will be shown. The requester can accept the stale results or ask QuiltView to obtain fresh results through one click in the web interface.

The demo will use one laptop to show web interface and three Google Glasses for participants to try on. We will also have three Galaxy Nexus phones that do screen-casting of Google Glasses so that the audience can see what the participants see through Glass. We will require Internet access and power for laptop and Glass charging.