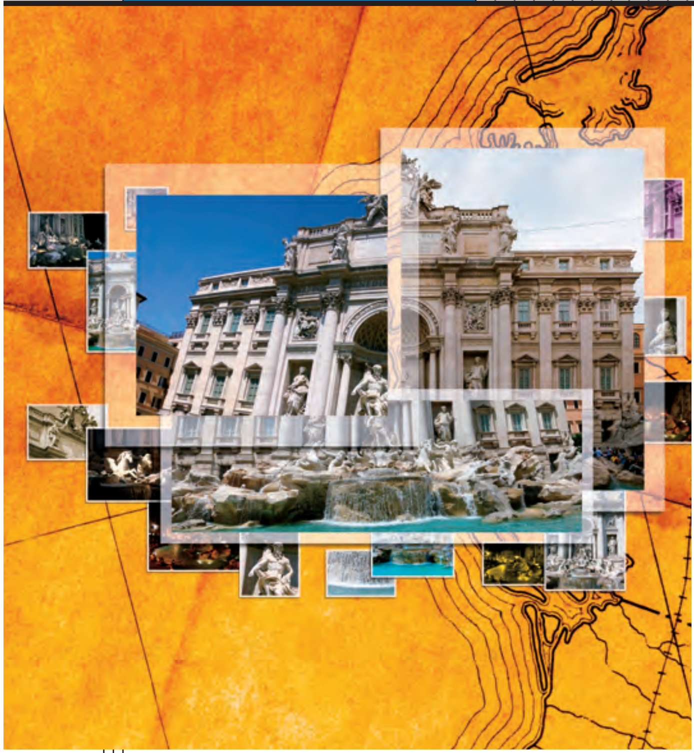

GOOGLE STREET VIEW: CAPTURING THE WORLD AT STREET LEVEL

Dragomir Anguelov, Carole Dulong, Daniel Filip, Christian Frueh, Stéphane Lafon, Richard Lyon, Abhijit Ogale, Luc Vincent, and Josh Weaver, *Google*

Street View serves millions of Google users daily with panoramic imagery captured in hundreds of cities in 20 countries across four continents. A team of Google researchers describes the technical challenges involved in capturing, processing, and serving street-level imagery on a global scale.

Several years ago, Google cofounder Larry Page drove around the San Francisco Bay Area and recorded several hours of video footage using a camcorder pointed at building facades. His motivation: Google's mission is to organize the world's information and make it universally accessible and useful, and this type of street-level imagery contains a huge amount of information. Larry's idea was to help bootstrap R&D by making this kind of imagery truly useful at a large scale. His passion and personal involvement led to a research collaboration with Stanford University called CityBlock (http://graphics.stanford.edu/projects/cityblock) that soon thereafter became Google Street View (www.google.com/streetview).

The project illustrates two principles that are central to the way Google operates. One is the hands-on, scrappy approach to starting new projects: instead of debating at length about the best way to do something, engineers and executives alike prefer to get going immediately and then iterate and refine the approach. The other principle is that Google primarily seeks to solve really large problems. Capturing, processing, and serving street-level imagery at a global scale definitely qualifies.

According to the CIA's *The World Factbook* (https://www.cia.gov/library/publications/the-world-factbook), the world contains roughly 50 million miles of roads, paved and unpaved, across 219 countries. Driving all these roads once would be equivalent to circumnavigating the globe 1,250 times—even for Google, this type of scale can be daunting. The Street View team has a long way to go to complete its mission, but by taking it one step at a time—developing sophisticated hardware, software, and operational processes, and relentlessly improving them—we've made good progress: Street View serves millions of Google users daily with panoramic imagery captured in hundreds of cities in 20 countries across four continents.

OPERATIONS OVERVIEW

Efficiently operating numerous data-collection vehicles around the world is a core problem the Street View team has had to address. True to Google's iterative approach to engineering, we developed and operated numerous ve-

Published by the IEEE Computer Society

hicular platforms in the project's four-year history. Figure 1 shows several of the platforms.

In Street View's "garage phase," the team mounted cameras, lasers, and a GPS on the roof of an SUV and placed several computers in its trunk. Enough imagery was captured to create demos, which were deemed sufficiently compelling to enter the next phase of growth.

Because we thought operations would dominate overall project cost, we designed a vehicle capable of capturing everything we might possibly need. The idea was to not have to drive anywhere more than once. A Chevy van was equipped with side- and front-facing laser scanners, two high-speed video cameras, eight high-resolution cameras in a rosette (R) configuration, and a rack of computers recording data to an array of 20 hard drives at 500 Mbytes per second. The van included special shock absorbers, a custom axle with rotary encoders used for pose optimization, and a heavy-duty alternator from a fire truck.

While this system enabled the team to collect initial data and develop end-to-end processing algorithms, it quickly became obvious that such a vehicle could not be built and operated at scale. We therefore quickly shifted to a third generation of vehicles, known as "lite" cars, this time focusing on off-the-shelf components and reliability, even at the cost of image quality. These bare-bones vehicles had a low-resolution camera connected to a standard desktop PC with a single hard drive. Instead of relying on custom axles, they recorded wheel encoder messages from the existing antilock brake system. These vehicles were quite successful, recording a vast amount of imagery in the US and enabling international expansion to Australia, New Zealand, and Japan.

This third-generation system's primary drawback was low image resolution. Thus, for the next-generation design, we developed a custom panoramic camera system dubbed R5. This system was mounted on a custom-hinged mast, allowing the camera to be retracted when the vehicle passed under low bridges. The R5 design also allowed us to mount three laser scanners on the mast, thereby enabling the capture of coarse 3D data alongside the imagery. This fourth generation of vehicles has captured the majority of imagery live in Street View today. A fifth-generation vehicle design is in the works.

In parallel with the road vehicles, the Street View team has developed numerous special data-collection platforms. We created the Trike to quickly record pedestrian routes in both urban and rural environments. It has been used everywhere from Legoland to Stonehenge, and is currently collecting data on UNESCO World Heritage sites. More recently, a snowmobile-based system was developed to capture the 2010 Winter Olympics sites in Vancouver. Aside from the obvious mechanical differences, it's essentially a Trike without laser scanners. This spring, snowmobiles are busily capturing imagery in numerous Colorado ski resorts.

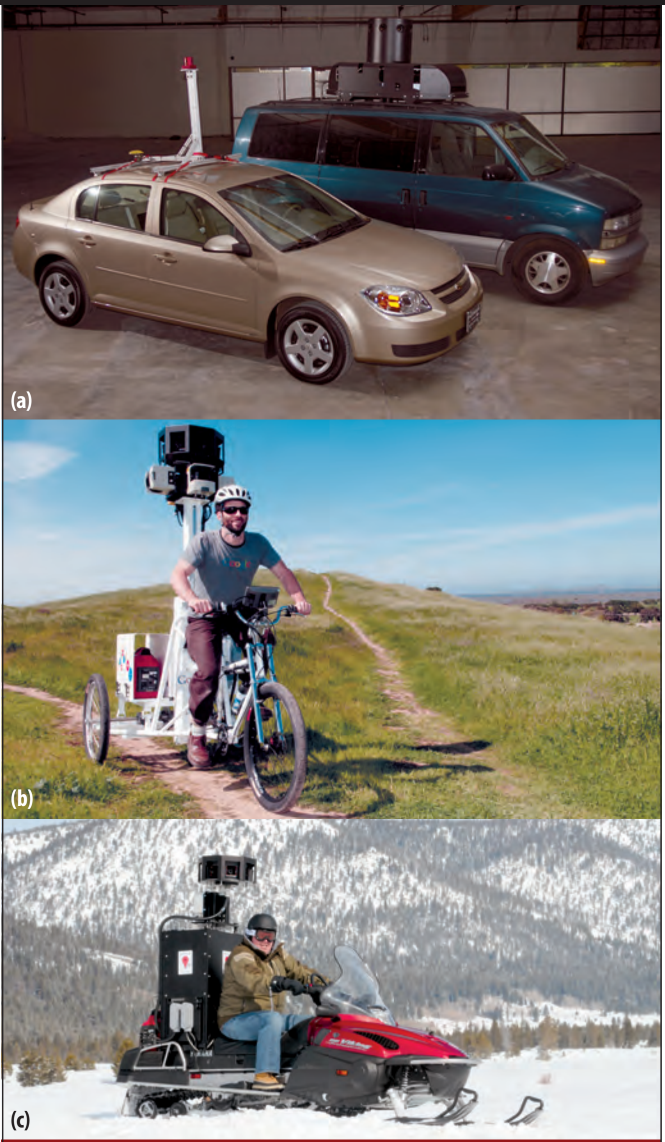

Figure 1. Street View vehicular platforms: (a) second- (right) and third- (left) generation custom road vehicles; (b) Trike; (c) modified snowmobile.

There are numerous challenges recording data using vehicular platforms. Hard drives are sensitive to shock, vibration, and temperature extremes, both while the vehicle is in operation and, to a lesser degree, while being shipped. We use various techniques to minimize data loss, from shock-mounted disk enclosures to custom-shipping packaging with extra-thick foam. Solid-state disk drives are also used when temperature and vibrations are expected to be extreme during data collection.

STREET VIEW CAMERAS

Street View imagery has come from several generations of camera systems. Most of the photos added in the past two years, primarily from outside the US, come from

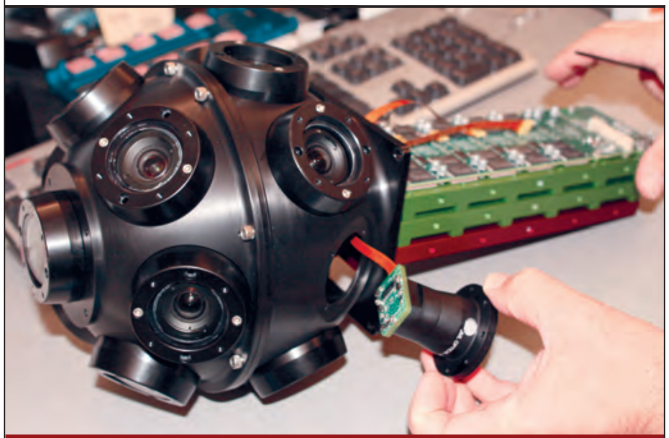

Figure 2. R7 Street View camera system. The system is a rosette (R) of 15 small, outward-looking cameras using 5-megapixel CMOS image sensors and custom, low-flare, controlled-distortion lenses.

the R5 system—the fifth generation of a series that we developed in-house, all of which provide high resolution compared to the "lite" vehicles' off-the-shelf camera. We're currently manufacturing and deploying its successor, R7.

Both R5 and R7 are rosettes of small, outward-looking cameras using 5-megapixel CMOS image sensors and custom, low-flare, controlled-distortion lenses. Some of our earliest photos were captured by R2, a ring of eight 11-megapixel, interline-transfer, charge-coupled device (CCD) sensors with commercial photographic wide-angle lenses. The R5 system uses a ring of eight cameras, like R2, plus a fish-eye lens on top to capture upper levels of buildings. Shown in Figure 2, R7 uses 15 of these same sensors and lenses, but no fish-eye, to get high-resolution images over an increased field of view—for example, to see down to sidewalks even on narrow streets. Other generations of rosette camera systems—R1, R3, R4, and R6—enjoyed a brief experimental existence but weren't deployed in quantity.

A big challenge of deploying cameras at scale is reliability, which we addressed through a ruggedized design, well-tested firmware and software, substantial vibration and temperature testing, burn-in tests of every rosette, and root-cause failure analysis. The deployed cameras have no moving parts, unlike some intermediate designs in which we experimented with mechanical shutters.

R2's interline-transfer CCD provides a global electronic shutter, but with this sensor architecture it's impossible to make the image quality robust for a camera looking into the sun. We tried mechanical shutters in R3 and R4 but settled on CMOS sensors with an electronic rolling shutter for R5 through R7. A key problem in these later designs was to minimize the distortion inherent in shooting from a moving vehicle while exposing each row of the image at a different time. The cameras must be in portrait orientation so that the exposure window's movement is roughly parallel to vehicle motion. If we sweep the exposure window from back to front, consistently across cameras, the main artifact is that foreground objects are distorted to appear thinner than they really are; this doesn't look bad compared to the kind of image-shear distortion that would be caused by operating in landscape orientation with the shutter window moving up or down.

POSE OPTIMIZATION

Accurate position estimates of Street View vehicles are essential for associating our high-resolution panoramas with a street map and for enabling an intuitive navigation experience. We use the GPS, wheel encoder, and inertial navigation sensor data logged by the vehicles to obtain these estimates. An online Kalman-filter-based algorithm is deployed on the vehicles to provide real-time navigation information to the drivers. As part of the Street View processing pipeline, we use a batch algorithm open sourced by Google (http://code.google.com/p/gpo/wiki/GPO) to achieve a smoother and locally accurate solution for the pose trajectory. This trajectory is computed at a resolution of 100 Hz, which is necessary to process laser data and to accurately correspond camera pixels to 3D rays in the presence of a rolling shutter.

Once the pose based on sensor fusion is computed, we align or snap the pose to a map of the road network. We construct a probabilistic graphical model of the network that represents all the known roads and intersections in the world. The model includes detailed knowledge about one-way streets and turn restrictions (such as no right turn or no U-turn). Using this model, we optimally transform the sensor pose into accurate road-positioning data. Among other things, this "road-based" pose permits us to provide navigation controls inside Street View to move along the road graph, display approximate street address information, and draw blue overlays on the map to indicate which roads have Street View coverage.

NAVIGATING STREET VIEW IMAGERY

Among the various ways Google Maps surfaces Street View images, the 360-degree panorama is probably the most popular. Several mobile and desktop clients leverage these images to produce an immersive experience in which the user can virtually explore streets and cities.

This experience becomes even richer as we combine Street View imagery with other data sources. For example, Street View is a powerful tool for finding local businesses, getting driving directions, or doing a real estate search. Building on the 3D data that we collect as well as Google Maps data, we can place markers and overlays in the scene, resulting in 3D-annotated Street View images.

Given Google's belief in empowering users, we recently opened Street View to user contributions. As Figure 3a shows, users can now correct the location of businesses

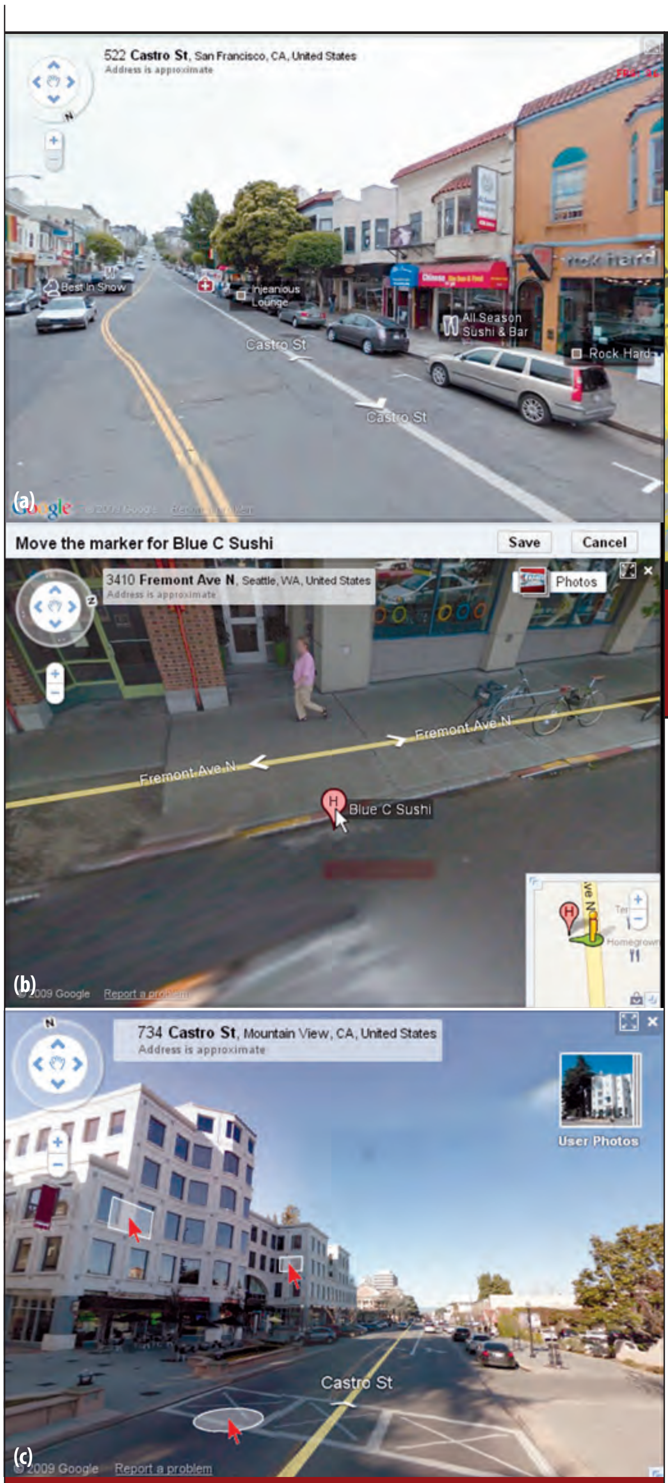

Figure 3. Navigating Street View imagery. (a) Users can correct the location of businesses and points of interest by directly dragging markers in Street View and automatically snapping them to facades. (b) They can also navigate from Street View pictures to matching user-contributed photos. (c) With the click-to-go feature, users can click their mouse on a point in the scene and be transported to the image nearest to that point's 3D location; if they hover their mouse's cursor over an image, they'll see a floating shape that shrinks in proportion to the depth and follows the underlying surface's normal geometry.

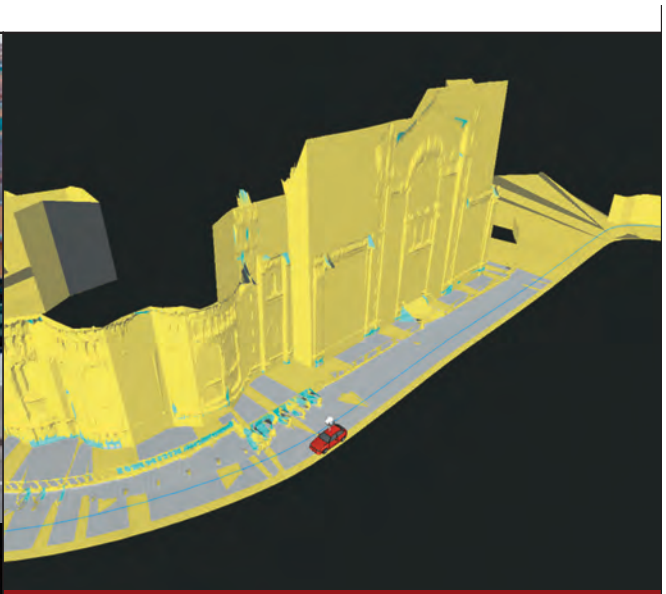

Figure 4. Imagery from new Street View vehicles is accompanied by laser range data, which is aggregated and simplified by robustly fitting it in a coarse mesh that models the dominant scene surfaces.

and points of interest by directly dragging markers in Street View and automatically snapping it to facades. With this ability, users can build map data with greater accuracy than ever before.

In addition to overlaying Street View imagery with local data, we also surface user-contributed photos from Flickr, Panoramio, and Picasa in Street View. As Figure 3b shows, users can navigate from Street View pictures to matching photos, and from there, directly within the file structure of user photos. This type of bridge between Google's images and user-contributed pictures is only a first step toward unifying navigation and browsing of georeferenced images.

LEVERAGING 3D DATA FOR SMART NAVIGATION

Street View supports a unique 3D navigation mode known as "click-to-go," which lets users click their mouse on a point in the scene and be transported to the image nearest to that point's 3D location. As Figure 3c shows, users can also hover the cursor over the image and see a floating shape that shrinks in proportion to the depth and follows the underlying surface's normal geometry.

Enabling such a feature requires the creation of a depth map that stores the distance and orientation of every point in the scene. As the imagery has a very high resolution, we compute a low-resolution depth map that can be quickly loaded over the network. For 3D navigation, we go further and create a depth map that only encodes the scene's dominant surfaces, such as building facades and roads, while ignoring smaller entities such as cars and people. We compute the depth of various points in the scene using

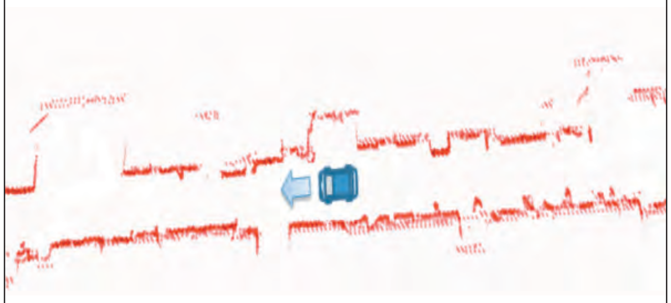

Figure 5. For imagery from older capture platforms, which lacks laser range data, depth is recovered by computing optical flow between successive images of the street facade on both sides of the vehicle. The optical flow at a given point depends on the vehicle's motion and that point's depth. To recover only the dominant scene surfaces, a piecewise planar global model of the facade is fitted to the optical-flow data over a long sequence of images.

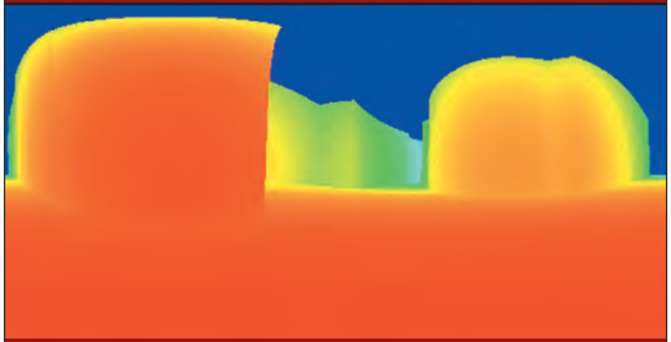

Figure 6. Depth map. Once the facade model is generated using lasers or computer vision, it's possible to render a panoramic depth map by tracing rays from each panorama position. Each pixel in the depth map represents a lookup into a table of 3D plane equations, which enables the client code to reconstruct the real depth values at runtime.

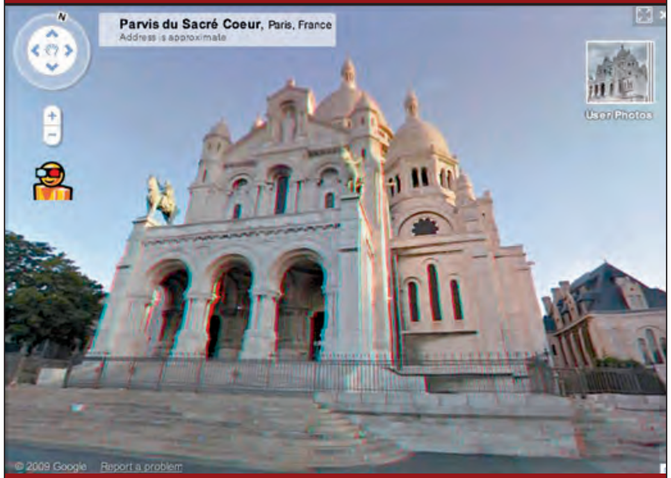

Figure 7. Panoramic 3D anaglyphs let users experience depth in Street View with simple red-cyan eyeglasses. Known depth is used to synthesize binocular parallax on the client side, thereby creating a second displaced view that replaces the red color channel in the original view.

laser range scans where available and image motion (optical flow) when laser data isn't available.

Imagery from new Street View vehicles is accompanied by laser range scans, which accurately measure the depth of a vertical fan of points on the two sides and the front of the vehicle. We aggregate this range data and simplify it by robustly fitting it in a coarse mesh that models the dominant scene surfaces. Figure 4 shows an example of a mesh simplified by plane fitting.

For imagery from older capture platforms, which lacks laser range data, we recover the depth by computing optical flow between successive images of the street facade on both sides of the vehicle. The optical flow at a given point depends on the vehicle's motion and that point's depth. To recover only the dominant scene surfaces, we enforce a piecewise planar global model of the facade and fit this to the optical-flow data over a long sequence of images. This process recovers the building facades and road geometry accurately and robustly. Our optimized depth estimation algorithm runs at about 50 frames per second on a contemporary desktop. Figure 5 shows an example of the recovered facade from a top view.

Once the facade model is generated using lasers or computer vision, we can render a panoramic depth map by tracing rays from each panorama position. In our implementation, each pixel in the depth map represents a lookup into a table of 3D plane equations, which enables the client code to reconstruct the real depth values at runtime. We further compact this representation using lossless compression. The encoded depth map is only a few kilobytes in size and can be quickly transported over the network to enable 3D navigation at the front end. Figure 6 shows an example of a depth map after decoding.

Recently, we've also used this depth map to synthesize panoramic 3D anaglyphs, letting users experience depth in Street View with simple red-cyan eyeglasses. Our approach uses the known depth to synthesize binocular parallax on the client side, thereby creating a second displaced view that replaces the red color channel in the original view to obtain an anaglyph, as shown in Figure 7.

COMPUTING 3D MODELS FROM LASER DATA

Taking extraction of 3D information even further, we use Street View data to create photorealistic 3D models for Google Earth. Traditionally, that service has created 3D city models from nadir or oblique airborne imagery, resulting in low-resolution facades with little detail—while suitable for fly-throughs, they didn't provide a pleasant walk-through experience. In contrast, 3D facade models reconstructed from Street View's laser scans and imagery are high resolution.

After filtering out noisy foreground objects, we synthesize a single consistent facade texture by aligning, blending, and mosaicking multiple individual camera images. To avoid duplication caused by multiple passes, and to create a single consistent facade model for an entire city, we resolve residual pose inaccuracies between acquisition runs and determine the most suitable set of final 3D facade models. We register these models with existing airborne models and then fuse them into a single model that has high-resolution facades as well as rooftops and back sides from an airborne view. Figure 8 shows how such a fused model enhances the user experience for a walk-through in New York City.

SERVING IMAGERY TO USERS

A launch pipeline processes the images captured by the Street View platforms to create user-visible imagery. We first stitch the images into panoramas and tile them at different zoom levels. The varying zoom levels enable clients to view the imagery according to the resolution of their viewports and to zoom in on some areas without having to download entire panoramas. We then run face detection and license-plate detection on all published imagery and blur images accordingly to protect privacy—a very compute-intensive step.[1]

Street View is currently available in more than 20 countries in Australia, Asia, Europe, North America, and Central America, with additional coverage under way. This represents a huge amount of imagery. To ensure low end-user latency, Street View servers are distributed in data centers around the world. Moreover, to ensure high availability while minimizing storage costs, we selectively replicate panoramas according to usage patterns.

Street View has been and continues to be an exciting adventure in global-scale photo collection, processing, and serving. The idea of driving along every street in the world taking pictures of all the buildings and roadsides seemed outlandish at first, but analysis showed that it was within reach of an organized effort at an affordable scale, over a period of years—at least in those parts of the world where political systems make it possible. The wide availability of street-level image data has proved to be very popular with users, delivering useful information that previously wasn't available. Looking forward, we continue to explore new interfaces, find better ways to integrate more user photos, map and photograph places such as malls and museums, and develop platforms to extend coverage to other places where cars can't go.

Reference

1. A. Frome et al., "Large-Scale Privacy Protection in Google Street View," *Proc. 12th IEEE Int'l Conf. Computer Vision* (ICCV 09), IEEE Press, 2009; http://research.google.com/archive/papers/cbprivacy_iccv09.pdf.

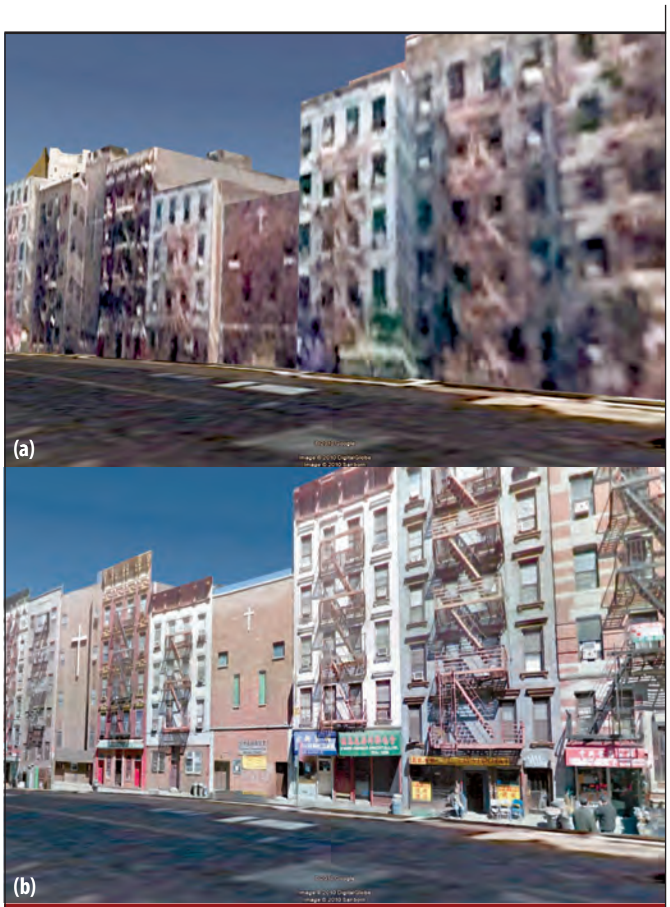

Figure 8. Using Street View data to enhance user walk-through experiences in Google Earth. (a) Original 3D models of a New York City scene from airborne data only. (b) Fused 3D model with high-resolution facades. The visual quality is considerably higher, and many storefronts and signs can now be easily identified and recognized.

Dragomir Anguelov *is a researcher working on pose estimation and 3D vision projects for Google Street View. He is interested in machine learning and its applications to computer vision, computer graphics, and robotics problems. Anguelov received a PhD in computer science from Stanford University. Contact him at dragomir@google.com.*

Carole Dulong *is a computer architect leading the Google Street View pipeline team. Her interests include application performance and large-scale parallelism. Dulong graduated as a computer science engineer from Institut Superieur d'Electronique de Paris. Contact her at caroled@google.com.*

Daniel Filip *is a computer scientist and manager working on 2D and 3D photo and panorama navigation for Google Street View, and was the first full-time Street View engineer. His research interests include computational geometry, computer vision, and computer graphics. Filip received an MS in computer science from the University of California, Berkeley. Contact him at daniel.filip@gmail.com.*

Christian Frueh *is a computer vision researcher leading 3D facade model reconstruction and model fusion efforts for Google Street View. His research interests include 3D modeling, laser scanning, and localization. Frueh received a PhD in electrical engineering after joint research at UC Berkeley and the University of Karlsruhe. Contact him at frueh@google.com.*

Stéphane Lafon *is a software engineer leading the front end and serving infrastructure for Google Street View. His research interests include numerical analysis, machine learning, and clustering. Lafon received a PhD in applied mathematics from Yale University. Contact him at stephane.lafon@gmail.com.*

Richard Lyon *is a research scientist who leads Google Street View's camera development team as well as a Google research project in machine hearing. His research interests include hearing, electronic photography, and biomimetic computing. Lyon received an MS in electrical engineering from Stanford University. He is an IEEE Fellow and a member of the ACM. Contact him at dicklyon@acm.org.*

Abhijit Ogale *is a computer vision researcher who works on extracting 3D information from Google Street View imagery. His research interests include optical flow, segmentation, 3D reconstruction, and action recognition. Ogale received a PhD in computer science from the University of Maryland, College Park. He is a member of the IEEE. Contact him at ogale@google.com.*

Luc Vincent *is the Google engineering director responsible for Street View. His research interests include image processing, document analysis, and computer vision. Vincent received a PhD in mathematical morphology from École Nationale Supérieure des Mines de Paris and an engineering degree from École Polytechnique, France. He is a member of the IEEE Computer Society. Contact him at luc@google.com.*

Josh Weaver *is a software engineer leading the vehicle and operational systems team for Google Street View. Weaver received an MS in electrical engineering and computer science from Massachusetts Institute of Technology. Contact him at jweaver@google.com.*