Photo Uncrop

Qi Shan[1], Brian Curless[1], Yasutaka Furukawa[2],
Carlos Hernandez[3], and Steven M. Seitz[1,3]

[1] University of Washington, Seattle, WA, USA
[2] Washington University in St. Louis, St. Louis, MO, USA
[3] Google Inc., Mountain View, CA, USA

Abstract. We address the problem of extending the field of view of a photo—an operation we call *uncrop*. Given a reference photograph to be uncropped, our approach selects, reprojects, and composites a subset of Internet imagery taken near the reference into a larger image around the reference using the underlying scene geometry. The proposed Markov Random Field based approach is capable of handling large Internet photo collections with arbitrary viewpoints, dramatic appearance variation, and complicated scene layout. We show results that are visually compelling on a wide range of real-world landmarks.

Keywords: Computational photography, image based rendering.

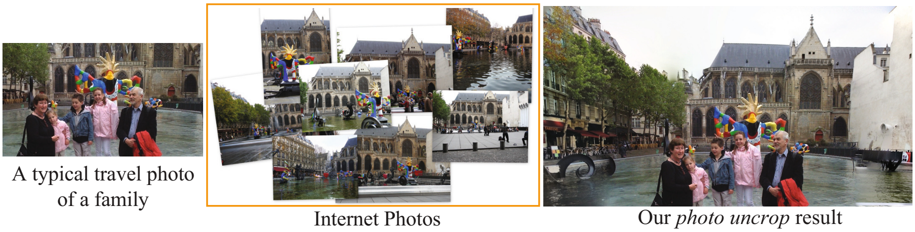
1 Introduction

Travel photos often fail to create the experience of re-visiting the scene, as most consumer cameras have limited field of view (FOV). Indeed, mobile phone cameras (which far outnumber any other photography device) typically have a FOV around 50-65 degrees, significantly narrower than the human eye [4]. Capturing large scenes is therefore tricky. Modern cell phones are equipped with camera apps providing a panorama mode, which allows you to take multiple pictures and stitch them into a bigger image. However, the process is often tedious. Furthermore, you cannot operate on your past photos. As a result, your photos are often more tightly *cropped* than desired (See Fig. 1).

We address the problem of extending the FOV of a photo—an operation we call *uncrop*. The goal is to produce a larger FOV image of the scene captured in your photo, leveraging other photos of the same scene from the Internet (captured at different times by other people). We make an important distinction between producing a *plausible* extended image using a technique such as texture synthesis [19], vs. producing an extended rendering of the *true scene* which is intended to be accurate. The latter case is more challenging and potentially more useful, as it gives you information about the real world, allowing you to *zoom out* of any photo to get better spatial context.

For almost any photo you take at a tourist site, there exist many other photos from nearby viewpoints, collectively capturing the scene across a potentially large FOV. Our approach is to automatically select, reproject, and composite a subset of this imagery into a large image screen centered on your photo. This problem is challenging for several reasons. First, the photos are not captured from the same optical center, resulting in too much parallax for existing state-of-the-art panorama stitchers (which produce severe

D. Fleet et al. (Eds.): ECCV 2014, Part VI, LNCS 8694, pp. 16–31, 2014.



A typical travel photo of a family

Internet Photos

Our *photo uncrop* result

Fig. 1. Capturing family photos with the desired background in the image frame can be tricky. Our approach expands the FOV of a user photo thus enables better spatial context. Landmark: Stravinsky Fountain in Paris.

artifacts as we will show). Second, the appearance (color, exposure, and illumination) varies dramatically between photos, making it difficult to produce a coherent composite. And finally, the presence of people, cars, trees, windows, and other transitory or hard-to-match objects make the alignment problem especially challenging.

This problem represents a compelling application that sits between traditional panorama stitching, which requires capturing many images and is thus labor intensive, and full 3D scene reconstruction, which has too many failure modes. Indeed, our experiments with state-of-the-art 3D reconstruction techniques [9,14,22] rarely produce hole-free geometry, omitting ground, people, trees, windows, and many other salient scene aspects. Our approach therefore assumes *incomplete* geometry in the form of depth maps, and leverages a novel Markov Random Field (MRF) based compositing technique to generate compelling full-scene composites complete with people, trees, etc. The method automatically generates results for multiple FOV expansions; the user can then choose the desired FOV and crop as desired to discard image boundaries with significant artifacts.

Our contributions are two-fold: (i) the first system to produce compelling uncropping results with dramatic boundary expansion from Internet photos; (ii) a novel MRF-based formulation adapted to handle significant geometry errors.

We show convincing results on a wide range scenes, each covered by 100s to 1000s of Internet images. Like existing panorama stitchers, our results are not entirely free of artifacts, and stitching seams and misregistration artifacts are occasionally noticable. However, we argue that for the intended application (giving you spatial context for your photo), small artifacts are quite tolerable. I.e., it's less important that every pixel is right than being able to zoom out and see that the building behind you is the Uffizi, or that you're standing in the middle of a large town square.

2 Related Work

Many texture synthesis techniques support image interpolation and extrapolation [19,28,13,5]; perhaps most related are those that leverage Internet imagery [24,11,15]. While these methods can produce extremely realistic results, they generally depict extrapolated scenes that don't actually exist; none of the extrapolation approaches attempt to capture the appearance of the real underlying scene.

There is a rich literature in panorama generation from multiple images sharing the same center of projection [23] with widespread popular deployment on smart phones [17]. There also exist large scale panorama creation projects, generating giga-pixel [16], and more recently tera-pixel [7] images.

When input images do not share the same center of projection, the alignment problem becomes significantly more difficult, as parallax, which depends on scene depth, must be taken into account. When parallax is small or for near-planar scenes, simple 2D image transformations such as homographies are often enough to align and blend images without artifacts [2,18,10].

In more general configurations, proper estimation of scene depths is essential for producing artifact-free images. Panorama stitching with scene depth estimation has been demonstrated for certain specialized camera motion cases including circles [23,26,21] and linear motion [20]. The addition of depth information enables new applications in these systems, such as the generation of depth of focus effects and 3D stereo images [21]. However, these techniques require continuous and often restricted camera paths and do not operate on community photo collections (e.g., Flickr) or other unstructured imagery. In this work, our goal is to extend the FOV of an input photograph by harnessing online community photo collections, via careful geometric analysis and blending techniques.

Most recently, and most similar to our own work, Zhang et al. [27] propose to expand the boundary of a personal photo (among other applications) using online collections. However, their method requires all images to overlap with the reference, limiting the effective expansion range. Further, they adopt a relatively simple, median-based averaging process for blending, which produces heavily blurred/ghosted composites on our examples.

An alternative approach would be to fully model geometry and reflectance of the scene, enabling (in principle) photorealistic scene rendering from any desired viewpoint. Despite exciting recent progress, however, state-of-the-art techniques rarely produce complete, high resolution reconstructions, and fail to model trees, people, windows, thin objects, and other very salient scene elements [22].

3 Input Data

We download images from Flickr (`http://www.flickr.com`) for a variety of sites, and use existing structure from motion (SfM) software [25] to compute camera poses. Uncropping is performed on images selected from the SfM model to show the capability of our system, though it would be straightforward to apply our system to an arbitrary new photograph by simply adding it to the relevant image set and performing incremental SfM. Publicly available multi-view stereo software is used to reconstruct per-view depthmaps [8]. Then, we warp each image by reprojecting its depth map and colors to the viewpoint of the image to be uncropped. More details on these preprocessing steps are found in Section 5.

4 Uncrop Algorithm

We propose an MRF-based compositing algorithm to construct a wide FOV target image around a reference image. We assign a label l to each source image, such that $l \in \{-1, 0, 1, \cdots, N-1\}$, where N is the number of images that survived the view selection process (including the reference image itself), and -1 is the null label. After re-projecting each source image, we have a set of partial, warped images $C_l(p)$ that each cover parts of the target image. We seek to solve for the label map $l(p)$ over target pixels p that will yield a high quality composite when copying warped image colors to the target image. We include the null label $l = -1$ to allow for a small number of pixels not covered by any of the images. After computing the composite, we perform a Poisson blend to give the final result.

We formulate the MRF problem as the sum of a unary term, a binary term, and a label cost term:

$$E(l) = \sum_p E_{\text{unary}}(p, l(p)) + \sum_{\{p,q\} \in \mathcal{N}(p,q)} E_{\text{binary}}(p, l(p), q, l(q)) + E_{\text{label}}(l). \qquad (1)$$

where $\mathcal{N}(p, q)$ denotes pairs of neighboring pixels in a standard 4-connected neighborhood. With abuse of notation, l here denotes the set of all the labels in the image. What is novel is the actual formulation of the unary and binary terms. We first describe their principles, where detailed formulation will be discussed in the following sections.

4.1 Principles

E_{unary}: It is nearly impossible to reconstruct perfect geometry for a complicated scene like ours, and a warped image may not be exactly aligned with the reference image. Therefore, the unary term incorporates the confidence of estimated depth information. Appearance mismatch is another source of artifacts. For example, compositing a daytime photo with a nighttime shot is challenging. We assign each image a score that measures the appearance similarity to the reference. Furthermore, appearance variation within an image due to shadows, over-saturation, and flash photography can result in spatially varying pixel quality. Thus, we assign lower cost to high contrast pixels.

E_{binary}: Traditional image stitching uses E_{binary} to minimize seams by looking for cuts on image edges. We follow a similar path, but also introduce a new measure to encourage any given reconstructed patch in the composite to resemble at least one warped source image at the same location. This helps to avoid making abrupt transitions in the composite that can arise from geometric misalignments, because noticeable artifacts at such transitions do not resemble corresponding regions in any of the input images.

E_{label}: Building a composite out of many images can lead to a quiltwork of stitched patches that can stray from the desired result. It is natural instead to encourage the stitcher to take pixel examples from a sparse set of warped views. In our approach, we achieve this by assigning a constant cost to each unique label used in the compositing.

4.2 Unary Term

We construct the unary term from several components:

$$E_{\text{unary}}(p,l) = E_{\text{geometry}}(p,l) + \alpha_1 E_{\text{appearance}}(l) + \alpha_2 E_{\text{contrast}}(p,l) + \alpha_3 E_{\text{reference}}(p,l), \quad (2)$$

where $\alpha_1 = 10$, $\alpha_2 = 5$, $\alpha_3 = 1$ are used in all of our experiments. Note that each warped source image $C_l(p)$ only partially covers the target image; if warped image l does not have a color at pixel p, the unary term is automatically set to infinity.

Geometry: We define the geometry term $E_{\text{geometry}}(p,l)$ as the possible error in the position of a reprojected pixel. It is determined by two factors: the accuracy of the original depth value and the baseline between the reference view and the source view. First, we model the accuracy using the range of depths in a local neighborhood in the source image l. More concretely, let u denote a source pixel in image l, and U to be the corresponding 3D point on the depthmap, which is re-projected to p in the reference. We look at a local neighborhood of size 11×11 pixels centered at u, and compute the minimum and the maximum depth values in the window. We have assumed a 1% depth error, and subtracts from the minimum and add to the maximum depth values by $0.01 D_u$, where D_u is the depth value at u. We take the 3D point U and shift its location to the minimum and the maximum depth locations, and project it to the reference image. Let us call the two projected location $p_{\text{near}}(p,l)$ and $p_{\text{far}}(p,l)$, respectively. Then, the geometry term is defined as follows:

$$E_{\text{geometry}}(p,l) = \max(|p - p_{\text{near}}(p,l)|, |p - p_{\text{far}}(p,l)|). \quad (3)$$

By minimizing this term, the optimization will favor pixels from images that have a smaller baseline relative to the reference view (less room for parallax errors) and images that sample surface regions more densely in close-ups and thus are more likely to cover a smaller range of depths. It is possible that multiple pixels u may warp to pixel p (see Section 5), in which case, we simply take the average projected location.

Appearance: Internet photos exhibit a wide range of illumination conditions. It is important to encourage the use of images with similar appearance. To do this, we assign an appearance cost to each source image. Specifically, we take the color histogram of each image, and score it by its KL divergence from the histogram of the reference image. Then the images are sorted in ascending order. Let k_l be the index of image l in this sorted list. We now define the overall image appearance cost as:

$$E_{\text{appearance}}(l) = k_l / N, \quad (4)$$

where N is the number of images in the set. Smaller cost in this case means less divergent from (more similar to) the reference image. Note that this unary term is constant for image l, regardless of which target pixel is being considered.

Contrast: Undesirable appearance variations such as shadows and over-saturation can be penalized based on the contrast. We address this by defining a local contrast cost. Let (G^l_x, G^l_y) be the finite difference gradient of image l after mapping image l to grayscale (intensity values $\in [0,1]$). We use the following formula to measure the lack of contrast over 11×11 window Ω centered at u in image l, which corresponds to p after the warping:

$$E_{\text{contrast}}(p,l) = \frac{1}{|\Omega|} \sum_{v \in \Omega} \sqrt{(1 - |G_x^l(v)|)^2 + (1 - |G_y^l(v)|)^2}. \tag{5}$$

If multiple pixels from source image l map to p after warping, we again simply take the average of their scores.

Reference: Finally, it is important to respect the reference image. Let us define the core region of the image Ω_{core} to be a set of pixels inside the reference image and more than 11 pixels in distance from its boundary. The reference cost is defined by applying the following four rules from top to bottom:

$$E_{\text{reference}}(p,l) = \begin{cases} 0, & l = l_{\text{ref}} \\ 10000, & l = -1 \\ 100, & p \notin \Omega_{\text{core}} \\ \infty, & p \in \Omega_{\text{core}} \end{cases} \tag{6}$$

where l_{ref} is the label of the reference image. It is possible that some of the pixels in the target image are not covered by any of the images, thus we allow the $l = -1$ label, with high cost.

4.3 Binary Term

Similar to previous work [3], we encourage label switches in regions with edges, where seams will be less noticeable. Further, we use a novel compatibility term to encourage constructing regions in the target image that resemble warped source image regions. Our binary term can then be written:

$$E_{\text{binary}} = E_{\text{edge}} + \beta E_{\text{compatibility}}. \tag{7}$$

where β trades off the relative contribution of the compatibility term. (We set $\beta = 10$ in all of our experiments.)

Edge: We first define a Sobel filter cost for a single pixel u and in (unwarped) source image l:

$$E_S(u,l) = \left(6 - \frac{||S(u,l)||_1}{4}\right)^2. \tag{8}$$

$S(u,l)$ is the concatenation of the Sobel filter responses in the x and y directions for each of the r, g, and b color channels, where we take the L_1 norm of this 6-dimensional vector. Now, for neighboring target pixels p and q with labels l and m, respectively, the binary edge cost is:

$$E_{\text{edge}}(p,l,q,m) = \begin{cases} 0, & l = m \\ E_S(u,l) + E_S(u,m), & l \neq m. \end{cases} \tag{9}$$

If multiple pixels correspond to p after warping, we take their average over u.

Compatibility: To encourage regions in the target image to resemble regions in the source image, we introduce a novel label compatibility term. Consider a pixel p and one of its neighbors q in the target image, and an image l. We define an 11×11 window around the two pixels and collect the pixels of $C_l(p)$ (corresponding to the warped version of image l) in the overlap into a vector $W_{p,q}(l)$. If there will be a transition between labels l and m in going from p to q, respectively, then the resulting window in the final result will likely resemble the average of the windows $W_{p,q}(l)$ and $W_{p,q}(m)$. This average in turn should resemble at least one of the (warped) source images. Thus, we define the following compatibility cost:

$$E_{\text{compatibility}}(p, l, q, m) = 1 - \max_n \text{NCC} \left[\frac{1}{2} \left(W_{p,q}(l) + W_{p,q}(m) \right), W_{p,q}(n) \right] \quad (10)$$

where $NCC[\cdot, \cdot] \in [-1, 1]$ is the normalized cross-correlation between two vectors, and n ranges over all of the labels. Note that, by this definition, this term becomes 0 when $l = m$. In addition, we set the term to ∞ if either $W_{p,q}(l)$ or $W_{p,q}(m)$ includes pixels where $C_l(p)$ or $C_m(p)$ are undefined.

4.4 Label Cost

We encourage the image stitcher to take color from a small number of images by assigning a constant cost for each additional label. If K is the number of unique labels in the composite, we set $E_{\text{label}}(l) = 500000 \cdot K$.

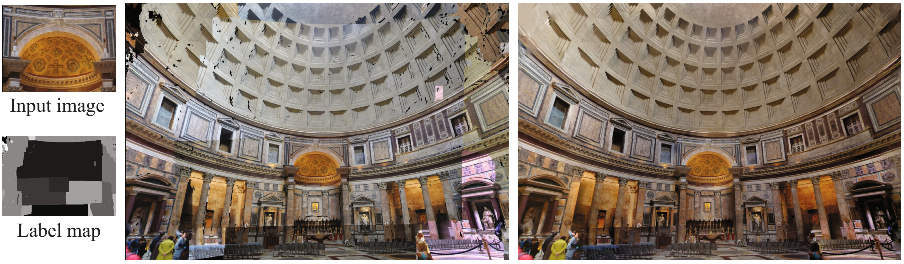
4.5 Optimizations and Accelerations

The energy definition in Eq. (1) falls naturally in the category of multi-label optimization with label cost. We optimize it with an iterative alpha-expansion solver [6].

Directly solving the problem is impractical due to the image resolution (millions of pixels) and the large label space (thousands of labels). Therefore, we apply (i) a simple up-sampling scheme and (2) a pre-filtering process to limit the solution space. The computational time varies from 10 seconds to a few minutes for solving the graph cut problem with a single thread on a 3.4Hz CPU.

Up-Sampling a Lower Resolution Label Map: The iterative alpha-expansion solver is performed on a target image that is $1/8$ the resolution (in each dimension) of the desired result. After optimization, the label values are upsampled as follows. Each pixel in the original high resolution target image has four possible label candidates at the 4 nearest pixels in the low-resolution label image. We simply pick the label with the lowest appearance penalty (Eq. 4).

Pre-filtering: First, we reduce the label set by discarding input images that are far from the COP of the reference view or cover only a small portion of the target image (see Section 5 for more details of this process). Next, we observe that the optimization process tends to reject pixels that (i) have large geometry cost, (ii) have poor patch compatibilities, or (iii) are too dark or over-saturated (essentially, pixels in solid black or white regions). Removing some obviously low quality pixels before performing the


Input image

Label map

MRF composite (without Poisson blending)

Final blend composite

Fig. 2. Landmark: Pantheon in Rome. Typically 10-20 unique labels are present in the label map after the graph-cut optimization. It is used to create an MRF composite.

optimization limits the solution space and can thus greatly improve the computational efficiency. Specifically, we remove a label l at pixel p from the solution space, that is, assigning infinity cost, when (i) $E_{\text{geometry}}(p, l) > 20$, $E_{compatibility}(p, l, p, l) > 0.6$, or $E_{\text{contrast}}(p, l) > \sqrt{2} - 0.01$.

4.6 Poisson Image Blending

The final, blended composite is computed from the MRF composite by solving a Poisson equation (Fig 2). We first compute the x, y gradient from the MRF composite, and set the values to be 0 at places where the label changes or where the label is -1. The blended composite should keep the color from the reference images; thus, we set a large weight (1000) to penalize differences from the reference image colors at the locations where reference pixels are available.

5 Implementation Details

Depth Map Reconstruction: We use publicly available multi-view stereo software [8] to reconstruct per-view depth maps, then apply cross bilateral filtering [12] for smoothing, as noise and high frequency geometric details often cause artifacts during image warping. The local window radius is 50 and the regularization parameter is 0.16 (suggested by the code of [12]). Note that we use the corresponding color image as the reference for the bilateral filtering. This process also helps in filling in missing depth values, where kernel weights are simply set to 0 for holes in an initial depth map. Finally, we compute a normal per pixel based on the depths.

Image Selection and Warping: Given a reference photograph and the SfM reconstruction, we first remove each source image with an optical center that is more than a distance τ_{COP} from the reference; we set $\tau_{\text{COP}} = 50$[1] in our experiments.

[1] The length of 1 unit in our 3D models is the distance between the first pair of images selected by VisualSfM. The pair is selected to have a large number of features in common while having a sufficiently large triangulation angle (greater than 4 degrees between their optical axes).

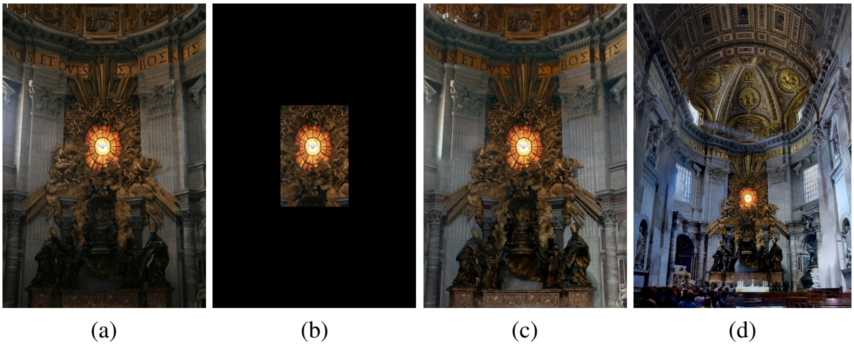


Fig. 3. Ground truth experiment (San Peter Cathedral). (a) The ground truth image. (b) We only keep $1/9$ of the image in the center, which is the input to our system. (c) Uncropped to the ground truth image size. The ground truth image in (a) was not used in creating this composite. (d) Uncropped to even wider FOV than the original.

Next, we forward-warp the remaining source images into the target image using splatting and a soft Z-buffer algorithm. We project each source image pixel into the target view, eliminating source pixels that are backfacing to the target view. In general, re-projected source pixels land between target pixels; furthermore, due to occlusions, foreshortening, and differences in image resolution, it is possible for multiple source pixels to land between the same set of target pixels. We associate each source pixel with the four nearest target pixels, storing at each target pixel p a sample $\{u, l, C, w, d\}$ comprised of the position u, image identifier l, color C, bilinear weight w, and re-projected depth d of the source pixel. We project all source images in this manner, storing a list of samples at each pixel. We then eliminate all samples that are behind the reference viewer ($d < 0$) or occluded by other samples based on a soft Z-buffer; i.e., for each target pixel p, we find the closest positive depth d_{closest} and consider a given sample with depth d at p to be occluded if $d > d_{\text{closest}} + \tau_{\text{depth}}$. (We set $\tau_{\text{depth}} = 20$ in our experiments.) For each target pixel p, we then collect all the samples from the same image l, compute a weighted average color $C_l(p)$ and a source pixel list $U_l(p)$, which will be used in computing label costs in the MRF formulation. Note that $C_l(p)$ only covers part of the target image and is "invalid" elsewhere; further, it is possible for source samples to land apart from each other due to grazing angle surfaces or if the source image is low resolution, leaving gaps between the projected samples.

Finally, we perform one last image selection step: for each image l, if the valid portion of $C_l(p)$ which lies outside of the reference image region covers less than 5% of the target image, then image l is eliminated from further consideration. This step tends to remove images that are: not looking in the direction of the scene of interest, are much lower resolution than the target image, or are close-ups of only a small portion of the scene of interest.

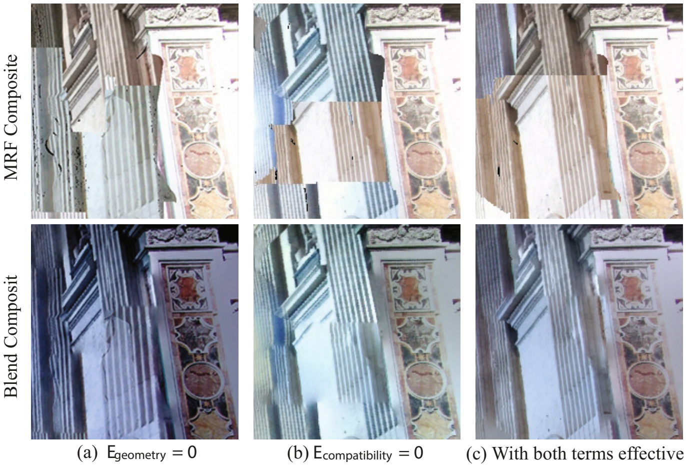

MRF Composite
Blend Composit
(a) $E_{\text{geometry}} = 0$
(b) $E_{\text{compatibility}} = 0$
(c) With both terms effective

Fig. 4. Evaluating the effectiveness of E_{geometry} and $E_{\text{binary compatibility}}$ (San Peter Cathedral). We show close-up views of the image mosaic and the Poisson blended results for better visualization. (a) E_{geometry} is turned off. (b) $E_{\text{binary compatibility}}$ is turned off. (c) Both terms are turned on.

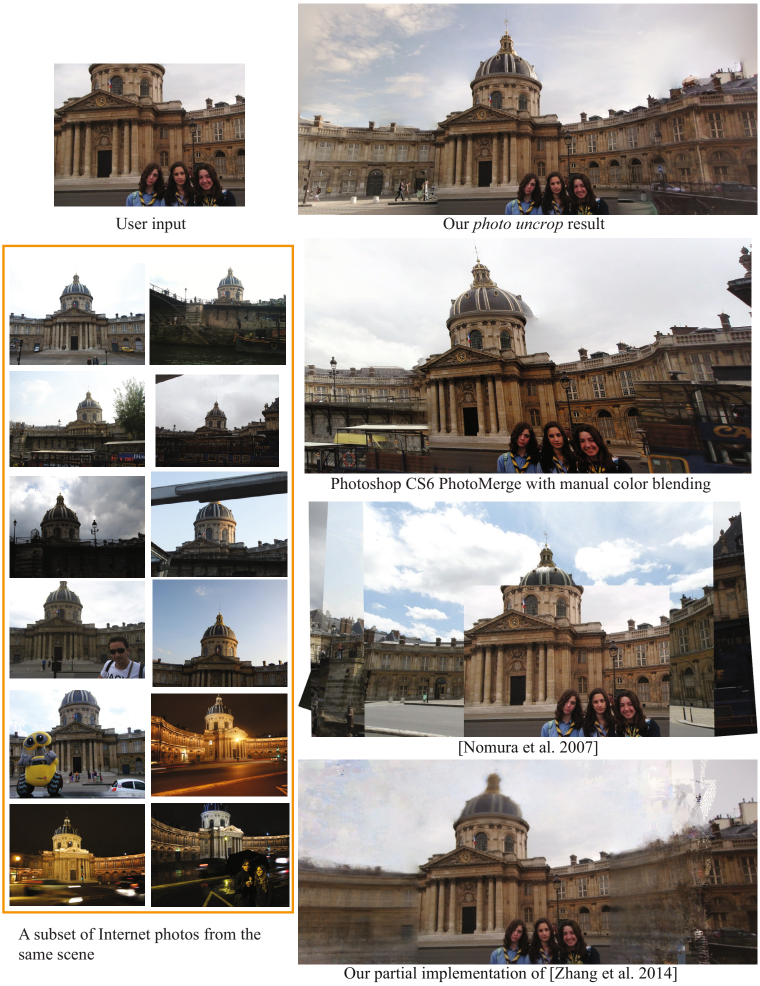
6 Results and Evaluations

We evaluated our system on 10 datasets from the city of Rome and Paris. The number of images in each dataset (i.e., SfM model) ranges from 262 (Stravinsky Fountain) to 2397 (Piazza Navona), where the largest two datasets contain more than 2000 images. We do not have enough space to show results on all the datasets, and refer the reader to the supplementary material[2] for more comprehensive results and evaluations. For each example, we generated results for several target image sizes and kept the largest image that looked plausible after manually cropping to discard image boundaries with significant artifacts. Automatically selecting the target image sizes and cropping is an area for future work.

6.1 Ground Truth Experiment

Figure 3 illustartes an experiment which allows us to compare our result againt the ground truth. We take a relatively wide FOV image (one from San Peter Cathedral dataset), crop to $1/9$ of the image in the center, then run our system to uncrop. Note that the ground truth image is not used for stitching. Despite minor intensity differences, our result faithfully reconstructs the original image using other photographs. In fact, our result has better contrast and reveals more details, in particular, in the bottom half of the image. To take this one step further, we can expand the FOV even more than the original image and generate a convincing composite with much wider field of view than the input.

[2] Please visit the project webpage at `http://grail.cs.washington.edu/projects/sq_photo_uncrop/` for more information.



User input

Our *photo uncrop* result

Photoshop CS6 PhotoMerge with manual color blending

[Nomura et al. 2007]

A subset of Internet photos from the same scene

Our partial implementation of [Zhang et al. 2014]

Fig. 5. Institut de France in Paris. We don't show the color blend result of [Nomura et al. 2007] since it is not straight forward from the output of their released executable.

6.2 Evaluation of the Geometry and Compatibility Terms

Here we evaluate the effectiveness of two novel components of our MRF formulation: E_{geometry} and $E_{\text{compatibility}}$. The E_{geometry} term prefers source pixels from smaller baseline views with more accurate depth estimates. These views typically produce fewer distortions. Fig. 4(a) shows the MRF composite and its Poisson blend when E_{geometry} is set to 0. The optimizer picks a patch with large geometry distortion, causing misalignment artifacts. On the other hand, $E_{\text{compatibility}}$ is designed to discourage switching labels to a misaligned image. We show the result of setting $E_{\text{compatibility}} = 0$ in Fig. 4(b). Severe misalignment is visible at the boundaries between image patches in the MRF composite. By incorporating both terms (Fig. 4(c)), the optimizer creates a better composite with fewer visible artifacts.

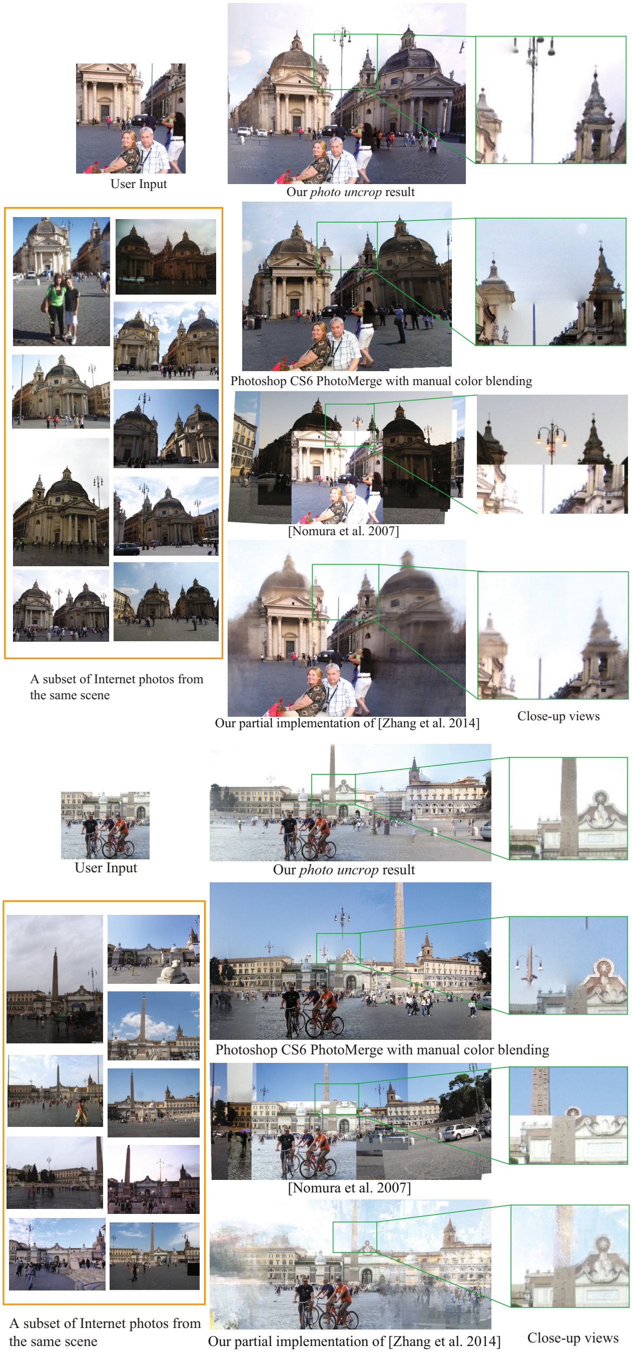
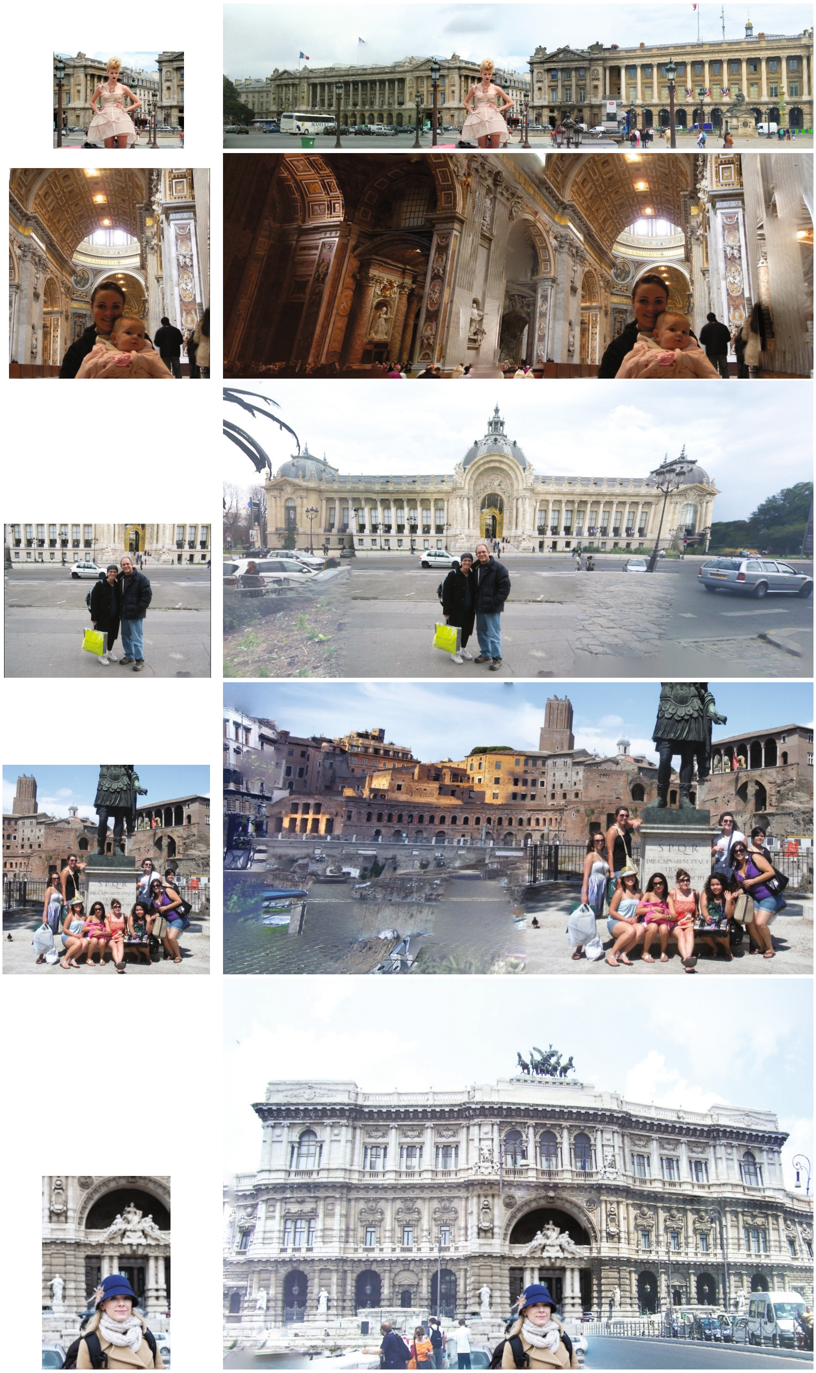

User Input
Our *photo uncrop* result
Photoshop CS6 PhotoMerge with manual color blending
[Nomura et al. 2007]
A subset of Internet photos from the same scene
Our partial implementation of [Zhang et al. 2014]
Close-up views
User Input
Our *photo uncrop* result
Photoshop CS6 PhotoMerge with manual color blending
[Nomura et al. 2007]
A subset of Internet photos from the same scene
Our partial implementation of [Zhang et al. 2014]
Close-up views

Fig. 6. Two datasets from Piazza del Popolo. Notice the geometry misalignment in results from PhotoMerge and [Nomura et al. 2007], as well as the blurred composites from [Zhang et al. 2014].



User input

Photo uncrop results

Fig. 7. More results

6.3 Comparitive Evaluation against Baseline Methods

To the best of our knowledge, there does not exist a system that can achieve unlimited FOV expansions on the same uncropping problem by chaining together overlapping community photos. The closest ones are the Photoshop CS6 PhotoMerge tool [1], Scene Collage [18] (with executables released), and the boundary expansion method in [27]. Here we treat the first two as baseline methods. Neither of them is capable of handling the large amount of images in our datasets (processes crash with our 64-bit Windows machine with 48 GB memory). To favor the baseline methods, we provide them with the set of images, which pass the pre-filtering process described in Sec. 5, where the number of remaining images is typically around 100. The source code for the third method [27] (which assumes that all images overlap the input) was not available and was not straightforward to reproduce: it involves many steps including depth-based warping in some areas, homography warping in others, texture synthesis in other parts, and seam carving for still other parts, and the description of the method is fairly brief and high-level. Instead, we used own warping method, which allows wider FOV expansion, and just applied the median-based blending step described in [27] to evaluate the compositing part of their pipeline.

A common problem of the baseline methods is the inability to handle non-planar geometry and reason about visibility, as shown in Fig. 5. Both PhotoMerge and Scene Collage copy pixels from a bridge that is *behind* the camera. The baseline methods usually prefer wider FOV source images, thus tend to use images containing occluders, the bridge and the bus in this case, in the composite.

The presence of large parallax is also a challenge for the baseline methods. Most 2D image transformations used for image stitching, such as a planar homography, are not sufficient to correctly warp images, unless the underlying geometry is near planar. This problem is well illustrated at the top portion of Institut de France in Fig. 5. Results in Fig. 6 show similar misalignment artifacts with the baseline methods, where our composites are significantly better.

Finally, for our examples, the simple median-based blending approach used in [27] produced heavily blurred/ghosted composites (Fig. 5, 6).

More experimental results are provided in Fig. 7, which clearly illustrates that the uncropped images with extended FOV provides better spatial context of the scenes.

7 Conclusion

This paper presents the first work on utilizing Internet imagery to extend the field of view of a user photo. We employ multi-view stereo to warp images into a target, wide FOV image and propose a novel MRF-based formulation designed to handle inevitable geometric inaccuracies. It creates results with image content that resembles the *real scene*. The evaluations on a wide range of real world datasets demonstrate the effectiveness of our approach. The results, while not perfect, are convincing and provide real spatial and visual context not available in the original user photo.

Our approach does have limitations. First, it only works for photos taken at sites where a sufficient number of Internet photos are available (e.g., tourist sites with 100s to 1000s of images in our examples) and would fail to reconstruct regions where there is no coverage. The ground is often a problem area, as people seldom photograph the ground (examples in Fig. 7). As with most panorama stitchers, transient objects in the source images – e.g., people and cars – can be problematic, and seams through them may occur. Recognition and segmentation algorithms could help address this problem. If the user photo itself contains transient objects that are not entirely in frame, then they will remain clipped in the final composite if the new field of view extends beyond them; automatically and realistically extending such objects (people, cars, etc.) out of frame would be interesting if quite challenging.

Acknowledgments. This work was supported by funding from National Science Foundation grant IIS-0963657, Google, Intel, Microsoft, and the UW Animation Research Labs.

References

1. Adobe: PhotoShop CS6 PhotoMerge, `http://helpx.adobe.com/en/photoshop/using/create-panoramic-images-photomerge.html`
2. Agarwala, A., Agrawala, M., Cohen, M., Salesin, D., Szeliski, R.: Photographing long scenes with multi-viewpoint panoramas. SIGGRAPH (2006)
3. Agarwala, A., Dontcheva, M., Agrawala, M., Drucker, S., Colburn, A., Curless, B., Salesin, D., Cohen, M.: Interactive digital photomontage. SIGGRAPH (2011)
4. Apple: iPhone 5 Specifications, `http://support.apple.com/kb/sp655`
5. Barnes, C., Shechtman, E., Finkelstein, A., Goldman, D.B.: PatchMatch: A randomized correspondence algorithm for structural image editing. SIGGRAPH 28(3) (2009)
6. Delong, A., Osokin, A., Isack, H.N., Boykov, Y.: Fast approximate energy minimization with label costs. IJCV 96(1), 1–27 (2012)
7. Fay, D., Fay, J., Hoppe, H., Poulain, C.: Terapixel, `http://research.microsoft.com/en-us/projects/terapixel/`
8. Fuhrmann, S., Goesele, M.: Fusion of depth maps with multiple scales. SIGGRAPH Asia (2011)
9. Furukawa, Y., Curless, B., Seitz, S.M., Szeliski, R.: Towards internet-scale multi-view stereo. In: CVPR (2010)
10. Garg, R., Seitz, S.M.: Dynamic mosaics. 3DimPVT (2012)
11. Hays, J., Efros, A.A.: Scene completion using millions of photographs. SIGGRAPH 26(3) (2007)
12. He, K., Sun, J., Tang, X.: Guided image filtering. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part I. LNCS, vol. 6311, pp. 1–14. Springer, Heidelberg (2010)
13. Hertzmann, A., Jacobs, C.E., Oliver, N., Curless, B., Salesin, D.H.: Image analogies. SIGGRAPH (2001)
14. Jancosek, M., Pajdla, T.: Multi-view reconstruction preserving weakly-supported surfaces. In: CVPR (2011)
15. Kaneva, B., Sivic, J., Torralba, A., Avidan, S., Freeman, W.T.: Infinite images: Creating and exploring a large photorealistic virtual space. In: Proceedings of the IEEE (2010)
16. Kopf, J., Uyttendaele, M., Deussen, O., Cohen, M.F.: Capturing and viewing gigapixel images. SIGGRAPH 26(43) (2007)
17. Microsoft: Photosynth, `http://photosynth.net/preview`
18. Nomura, Y., Zhang, L., Nayar, S.: Scene collages and flexible camera arrays. In: Eurographics Symposium on Rendering (2007)

19. Pritch, Y., Kav-Venaki, E., Peleg, S.: Shift-map image editing. In: ICCV (2009)
20. Rav-Acha, A., Engel, G., Peleg, S.: Minimal aspect distortion (MAD) mosaicing of long scenes. IJCV 78(2-3), 187–206 (2008)
21. Richardt, C., Pritch, Y., Zimmer, H., Sorkine-Hornung, A.: Megastereo: Constructing high resolution stereo panoramas. In: CVPR (2013)
22. Shan, Q., Adams, R., Curless, B., Furukawa, Y., Seitz, S.M.: The visual Turing test for scene reconstruction. In: Joint 3DIM/3DPVT Conference (3DV) (2013)
23. Shum, H.Y., Szeliski, R.: Stereo reconstruction from multiperspective panoramas. In: ICCV (1999)
24. Whyte, O., Sivic, J., Zisserman, A.: Get out of my picture! internet-based inpainting. In: Proceedings of the 20th British Machine Vision Conference, London (2009)
25. Wu, C.: VisualSFM: A visual structure from motion system, `http://ccwu.me/vsfm/`
26. Zelnik-Manor, L., Peters, G., Perona, P.: Squaring the circle in panoramas. In: ICCV (2005)
27. Zhang, C., Gao, J., Wang, O., Georgel, P., Yang, R., Davis, J., Frahm, J.M., Pollefeys, M.: Personal photo enhancement using internet photo collections. TVCG (2014)
28. Zhang, Y., Xiao, J., Hays, J., Tan, P.: Framebreak: Dramatic image extrapolation by guided shift-maps. In: CVPR (2013)