

Object Removal in Panoramic Media

Andrew MacQuarrie, Anthony Steed
Department of Computer Science, University College London
a.macquarrie@cs.ucl.ac.uk, a.steed@ucl.ac.uk

ABSTRACT

Due in large part to new consumer virtual reality systems, panoramic media is an increasingly popular image and video format. While the capture of panoramic media is well understood, editing still poses many challenges. In this paper we explore object removal in 360° images. First, a method is proposed in which field-of-view expansion using retargeting techniques is combined with Graphcut Textures to remove objects near the equator of the viewing sphere. Several extensions and refinements are proposed to improve this technique, including how it can be extended to removing objects anywhere on the viewing sphere. Secondly, inpainting in 360° images is examined, with an exploration of how the choice of projection affects the inpainting result. Finally, the latter technique is shown to work for video in certain situations.

Categories and Subject Descriptors

I.4 [Computing Methodologies]: Image processing and computer vision; I.4.9 [Computing Methodologies]: Image processing and computer vision—*Applications*

Keywords

Computational photography, Image manipulation, Panorama images

1. INTRODUCTION

Cinematic virtual reality is the creation and display of video that has a large field of view (FOV), covering up to a full-sphere around the viewer of 360° horizontally and 180° vertically. This is often referred to as the *viewing sphere*. While cinematic virtual reality is not a new concept, the field has seen a rapid increase in interest over the last few years. This rise has been driven by the falling cost and rising quality of capture and playback devices. Mass-produced, wide-angle “action cameras”, such as the GoPro HERO, have brought the cost of building a 360° camera rig down significantly, while an array of purpose-build devices are also entering the market [13]. Playback support has been integrated into YouTube, allowing easy viewing in browsers and phones [10]. Meanwhile,

head-mounted displays such as the Oculus Rift and Google Cardboard have brought immersive, panoramic media playback to the mainstream.

Although it remains a practical challenge for filmographers, the capture of 360° media is now well understood. Editing panoramic media, however, brings new challenges. Due to the fact that filming without the need for post-production is often prohibitively difficult, the editing of images and video has been studied extensively. This research has resulted in the development of many algorithms that produce excellent results in areas such as object removal, hole filling, retargeting and reshuffling. However, it is not yet clear what must be done to allow these algorithms to produce equally good results on panoramic media. Key differences – such as the projection that spherical content must undergo in order to be edited and stored effectively – strongly indicate that these algorithms cannot be applied without alteration. Additionally, there are likely improvements which can be made by adapting these techniques to make use of the properties of panoramic media, such as the wealth of additional information that is captured over and above that of a regular camera.

In this paper, we consider a problem frequently experienced during panoramic media production – object removal. As full-sphere cameras capture all directions at once, any equipment or crew that are in the line of sight of the camera will be captured. Entirely manual object removal is labour intensive and expensive. By adapting existing semi-automatic object removal algorithms to work for panoramic content, it is hoped that more effective post-production can be used to reduce the cost of 360° media. First, a method of object removal in 360° images is proposed, in which field-of-view expansion is combined with Graphcut Textures to remove objects. Second, inpainting in 360° images is examined. Finally, the latter technique is shown to work for video in certain situations.

2. BACKGROUND

2.1 Panorama stitching

Cinematic virtual reality content is usually filmed using several cameras capturing overlapping views that are stitched together in software. Each view is warped into alignment using a homographic transformation, corrected for lens distortion and blended together using image processing [20][21][17][4]. Playback is then achieved using software or specialised hardware. Software viewers are available for desktops and mobile phones. For immersive viewing experiences, head-mounted displays can be used that allow the viewer to look in any direction naturally by turning their head.

2.2 Projections

While image processing operations could be done on the sphere, it is often simpler for the editing and storage of panoramic media to project this sphere onto a plane. Projecting a sphere onto a plane is not an easy task. Thankfully, the problem has been extensively studied for thousands of years due to the need to create 2D maps of the earth [19]. Through the study of cartography, a large number of different types of map projection have been designed.

The projection of a sphere onto a plane inherently introduces distortion. As a result, each type of map projection seeks to preserve some aspect of the original content. The simplest projection is the equirectangular projection. This projection maps the lines of longitude to vertical straight lines and the lines of latitude to horizontal straight lines [18]. A special case of the equirectangular projection is the plate carrée projection, in which the equator is the undistorted latitude. As the plate carrée is very common, it is usually referred to simply as “equirectangular” – a convention that this paper also follows. In this projection, the x coordinate of the image maps directly to the longitude and the y coordinate maps directly to the latitude. As a full-sphere panorama has a horizontal FOV (HFOV) of 360° and a vertical FOV (VFOV) of 180° , the equirectangular projection results in a rectangular image with a 2:1 aspect ratio.

Due to the nature of the equirectangular projection, there is very little distortion around the equator but substantial distortion at the poles. Equirectangular projections are not often used in cartography because of this large pole distortion. However, due to their simplicity and the ease with which scenes can be understood, equirectangular is probably the most commonly used projection in cinematic VR. Other notable projections include: Mercator, which preserves shapes locally; Sinusoidal, which preserves relative areas; and rectilinear (perspective), which preserves straight lines but introduces stretching distortions at the edge of the image when covering a FOV above 40° [5].

2.3 Graphcut textures

In section 3 it will become necessary to cut two images together in a way that disguises a join. Combining two images together in a plausible way can be achieved using a technique known as Graphcut Textures [9]. Two images are overlaid at their desired positions. A cut between them is then identified with the intention of disguising the join. This cut is found using a min-cut/max-flow optimisation.

To perform the min-cut/max-flow optimisation, a graph is constructed. Each pixel is connected to its neighbours, with weights favouring cuts at areas in the original images that appear similar. The weight for the arc between adjacent pixels s and t is defined by examining the surrounding pixels in the two images, using the formula:

$$M(s, t, \mathbf{A}, \mathbf{B}) = \|\mathbf{A}(s) - \mathbf{B}(s)\| + \|\mathbf{A}(t) - \mathbf{B}(t)\|$$

where $\mathbf{A}(s)$ is the patch from image \mathbf{A} at pixel s , $\mathbf{B}(t)$ is the patch from image \mathbf{B} at pixel t , and $\|\cdot\|$ denotes an appropriate norm. Arcs with infinite weight (“constraint arcs”) are used to ensure certain pixels are taken from a specific image. For example, for images \mathbf{A} and \mathbf{B} placed side by side and being joined by a vertical cut, pixels at the left hand side and right hand side of the overlapping area are taken from image \mathbf{A} and \mathbf{B} respectively. Min-cut/max-flow is then used to find an optimal seam. This seam identifies which image each of the unconstrained pixels should be copied from. Pixels to the left of the cut are taken from image \mathbf{A} while pixels to the right of the cut are taken from image \mathbf{B} . This algorithm can produce compelling results, particularly in scenes with similar areas in both images that allow a good cut.

2.4 Retargeting

As will be shown in section 3, it can be useful to alter the FOV of a displayed panorama. This can be different from the FOV a panorama captures. For example, a panorama that captures 360° HFOV could be warped to be displayed over only 180° . This can be achieved by altering the height and width of some projections, including equirectangular. Altering the width and height of media is referred to here as *retargeting*.

Due to the heterogeneous nature of display devices, retargeting media is a well studied area. The simplest method is to resize the content equally using a scaling algorithm such as bicubic interpolation [8]. Algorithms that aim to improve on this generally attempt to minimise noticeable distortion to salient content.

Seam carving is a method that repeatedly adds or removes pixel-wide, connected horizontal or vertical seams until the image conforms to the desired size [2]. These seams are chosen using some energy function – such as avoiding cuts in high-energy or salient areas – and a solution is found greedily per seam using dynamic programming. Other retargeting methods include non-homogeneous warping [23], and shift-maps which discretize the problem into the rearrangement of pixel positions [14].

2.5 Inpainting

In section 5, we will investigate inpainting in panoramic content. Inpainting is the filling of holes in a plausible way. It is a powerful technique that has gained extensive popularity. While other forms of inpainting such as shift-maps have been proposed [14], the technique most generally used is a patch-based method derived from previous work on texture synthesis [7][6]. Holes are filled from the outside in, propagating structure and texture by copying suitable content from elsewhere in the image or video, using a patch-based similarity measure to find a nearest-neighbour field (NNF).

The construction of the NNF presents complexity issues if approached in a brute force fashion. However, several excellent improvements have been proposed that calculate approximate nearest-neighbour fields (ANNF) at a fraction of the complexity. Patch-Match was a seminal work in this area [3]. It uses the coherence of images to propagate good matches, found via random sampling, to a pixel’s neighbours.

In this paper we make use of Adobe Photoshop’s content-aware fill tool. This is an excellent image inpainting implementation, that uses a patch-based hole filling method based on Space-time Video Completion, originally proposed by Wexler et al. [1][22]. Patch-Match is used to create the ANNF, which allows interactive speeds to be achieved.

2.6 Working with panoramic media

Sacht et al. have done work in panoramic media in the context of straight line and face detection [16]. They reached a similar conclusion to us - that working in panoramas requires finding a suitable projection for the job at hand. They discovered that the local shape preservation properties of the Mercator projection facilitates the identification of faces, while the perspective projection is required for line detection. Our contribution is the investigation of object removal in panoramic media. We propose a novel solution to deal with unwanted objects and holes in panoramic media, as well as investigating inpainting techniques.

3. FOV EXPANSION AND GRAPH CUTS

Object removal in a 360° panorama can be formulated as FOV expansion and cutting. By expanding the HFOV beyond 360° , an area of overlap is introduced to the sphere. If the overlap is posi-



(a) Equirectangular with object to remove in red.



(b) Section A is removed, while sections B and C are overlapped and a good transition found using Graphcut Textures.

“Sherbrooke Forest” by Peter Gawthrop, used under CC BY-NC 2.0

Figure 1: FOV expansion and Graphcut Textures

tioned at the point of the unwanted object, a cut between these ends can be done in such a way as to remove the object.

3.1 Algorithm formulation

This algorithm can be formulated using an equirectangular projection. This projection has the useful property that increasing the HFOV can be achieved by simply increasing the image width. For every $1/360\%$ increase to the width, we increase the HFOV by 1° .

An example of this method is shown in figure 1. The section of the equirectangular image containing the object is removed. As equirectangular images have a distortion pattern that is constant horizontally but varies vertically, the removal is performed vertically to ensure content maintains the correct variation of vertical distortion. At this stage, the image could be stretched to restore the 2:1 aspect ratio and considered a 360° panoramic image. However, the introduced cut would be highly noticeable and jarring. To disguise the cut, Graphcut Textures can be used.

To create a plausible join, the left hand side and right hand side of the join are overlapped as shown in figure 1b. A larger overlapping region provides more scope for a good cut to be found, however it also removes more of the original panorama. This is a parameter that can be altered depending on the context. Graphcut Textures is then used to find a good join between these overlapping sections, formulating the problem as a min-cut/max-flow optimisation that favours cuts at areas of similarity.

3.2 Results

We tested the above technique on equirectangular images. As can be seen in figure 1a, we digitally added an object to the scene that presented a substantial challenge to remove. We performed the technique as described in section 3.1, the results of which can be seen in figure 2.

The results are fairly compelling. The offending object is completely removed, and the cut is well disguised as can be seen in figure 2c. Some artifacts can be seen, for example the branch at the centre top of image 2c ends abruptly where the cut has taken place. We have also needed to remove more of the original panorama than in the copped version seen in figure 2b to allow an overlapping region for graph cuts to take place.

Two further examples can be seen in figure 3. Here, frames from two videos are processed to remove crew and equipment. The removal in figure 3c is challenging as the object is fairly wide in the scene, and the background change is nontrivial. To produce this cut, the section marked to be removed included the crew and most of the bag – the remainder of the bag was removed automatically during the graph cut phase. Importantly, these cuts take place in non-salient areas of 360° panoramas. This means that the viewer is most likely not focusing their attention in this direction, and therefore some minor artifacts may go unnoticed.

3.3 Limitations

By removing a section of the panorama and stretching the remaining content to restore the 2:1 aspect ratio, a circular distortion centred around the poles has been introduced. This circular distortion is noticeable even in natural scenes, as shown in figure 4. The amount of the panorama removed will affect the results, with smaller cuts introducing less distortion. The distortion effect in figure 4 is quite pronounced as 25% of the HFOV of the original panorama has been removed.

This distortion effect alters the content in such a way that straight lines are no longer guaranteed to be straight. This is particularly pronounced near the poles. This effect may or may not be noticeable depending on the content; natural scenes such as that of figure 1a may appear plausible, while artificial structures with straight lines will make this effect more obvious, as shown in figure 5. As we will discuss further in section 4.1, this effect can be reduced by scaling the content in a non-homogeneous fashion.

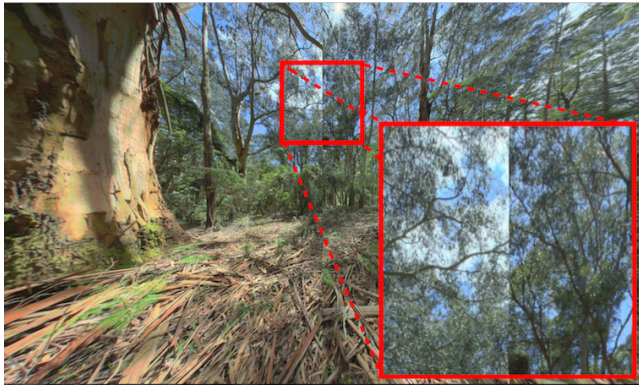
Marking the region to remove is very easy. Only the x-axis start and end positions of the region to remove, as well as the overlap size, need to be specified. This makes the process quick compared to the creation of even a rough mask. However, this speed comes at the expense of control. The entire marked region will be removed and the overlap searched for a good cut – the user cannot specify regions in the overlapping area they would like to keep. As the system takes only a few seconds to create the new panorama, a trial and error approach can be used to identify good settings for an image interactively.

The graph cut method used suffers the same limitations as Graphcut Textures. Sometimes no decent cut is available, and the two sides cannot be realistically combined. This can happen when a significant change happens in the background across the object to be removed. Additionally, artificial structures and buildings are difficult to cut plausibly as they tend to have a regular shape with many straight lines, which are challenging to join realistically. An example of this can be seen in figure 6. For this reason outdoor, natural scenes are likely to produce better results.

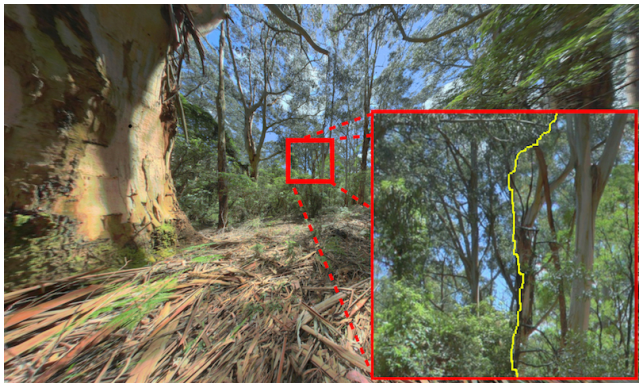
A cut alters the physical layout of the space. In situations where the viewer does not have a strong mental model of this layout, this is usually not an issue. In some cases, however, such as a square room that is no longer square following the cut, the alteration is quite pronounced and slightly disconcerting. In scenes with very regular surrounding structure, the technique of section 5 may well be applicable.



(a) Rectilinear view with object to remove in red.



(b) A simple straight cut: the cut appears as a noticeable sharp edge.



(c) Our graph cuts method: using Graphcut Textures, the cut is well disguised. In the closeup the cut is shown in yellow.

“Sherbrooke Forest” by Peter Gawthrop, used under CC BY-NC 2.0

Figure 2: Rectilinear views of image 1a following HFOV expansion and region removal. As a large FOV has been used to display more of the scene, stretching distortion can be seen at edges.



(a) Before cut: the director is visible and should be removed. (b) After cut: the seam is not obvious, but minor artifacts in shadows. Frame courtesy of Peter Boyd Maclean



(c) Before cut: some crew and equipment are visible. (d) After cut: minor artifacts are noticeable in the sky. Frame courtesy of the BBC

Figure 3: Rectilinear views of equirectangular video frames, showing object removal using our graph cuts method.



(a) Before cut: the south pole appears normal. (b) After cut: a circular distortion has been introduced, centred on the pole.

“Sherbrooke Forest” by Peter Gawthrop, used under CC BY-NC 2.0

Figure 4: Rectilinear views of the circular distortion introduced by our graph cuts method at the south pole.



(a) Before cut: straight lines near the north pole appear straight. (b) After cut: straight lines near the north pole appear curved.

Figure 5: Rectilinear views of the tops of buildings, showing distortion introduced at the north pole by our graph cuts method.



(a) Before cut: the person in the red coat should be removed. (b) After cut: background structures are noticeably incorrect.



(c) Before cut: the teleprompter should be removed. (d) After cut: a good cut cannot be found.

Frame courtesy of the BBC

Figure 6: Failure cases of our method: rectilinear views before and after cuts. The objects are successfully removed, however surrounding structures are incorrect resulting in unrealistic images.

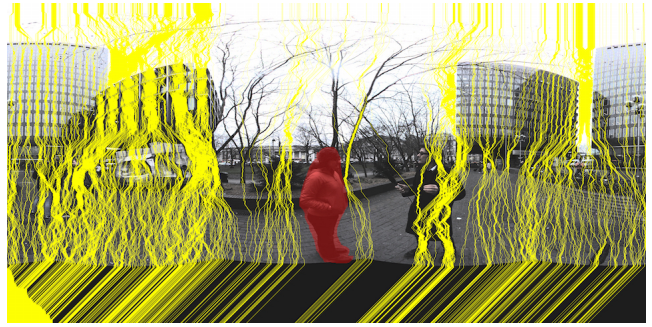
4. EXTENSIONS & REFINEMENTS

4.1 Retargeting techniques

The method described in section 3 performs homogeneous stretching of content to restore the 2:1 aspect ratio of the equirectangular image following a cut. While this can produce good results, it can also cause noticeable distortion. For example, following the removal of a large object, people are visibly more stretched in the horizontal axis, and buildings with straight lines close to the poles no longer appear straight. Retargeting techniques can be used to reduce these distortions.

Seam-carving can be applied to prevent the distortion of important sections. These sections, for example people or straight lines near the poles, could be detected automatically in panoramic images using the methods proposed by Sacht et al. [16]. However for simplicity, the example presented here was produced using a manually created saliency mask. As shown in figure 7, additive seam carving was used to increase the width of an image while preserving the salient element, in this case a person who is expected to be the focal point.

Seam carving in panoramic content may be even more effective than in standard format media due to the potentially large total area of the image that is non-salient. Standard media has a comparatively small FOV, so often the entire screen is filled with salient content. Panoramic media, in contrast, generally has a large amount of non-salient content. However, seam carving is a computationally expensive process in this context due to the extremely large size of panoramic images. Carving 600 seams in the moderately sized 2400x1500 frame shown in figure 7a took 15 minutes, despite the



(a) Equirectangular undergoing seam carving to restore 2:1 aspect ratio following large object removal. Saliency mask is highlighted in red, carved seams are shown in yellow.



(b) Rectilinear view: (c) After cut: homogeneous image stretch. (d) Seam carved: person preserved by mask.

Figure 7: Seam carving can be used to preserve salient content during FOV expansion.

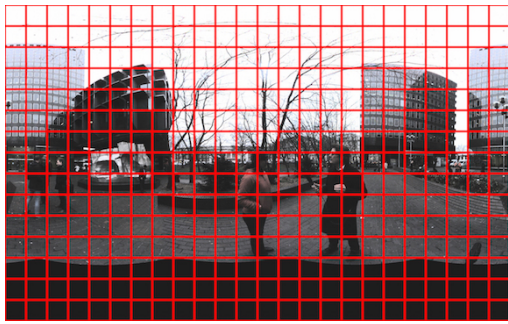
use of a well known, fast implementation [15]. Additionally, seam carving can introduce noticeable distortions in some scenes, particularly regular structures and straight lines. Instead, a simpler method can be used.

By performing stretching of the non-salient content only, good results can be achieved at interactive speeds. After manually determining the salient areas, non-salient content is stretched horizontally to restore the 2:1 aspect ratio following a cut, as can be seen in figure 8. As we are only stretching along the x-axis and a large amount of non-salient content is available to perform the operation easily, a very simply method can be used. The sections to the left and right of the salient area are stretched, while the central region containing the salient content retains its original size. Similar results to those of figure 7 are achieved, in less time and with fewer distortions introduced in areas with regular structure. If a more complex scaling was required, for example in scenes with many disconnected salient areas, a more appropriate stretch may be facilitated by a non-homogeneous image warping algorithm; an example of such an algorithm is that proposed by Wolf et al., which also allows the automatic, saliency-based warping of video [23].

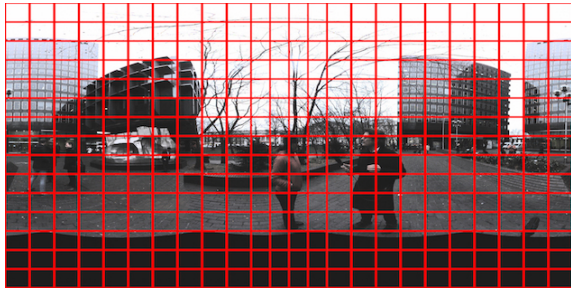
4.2 Tripod removal

The method outlined in section 3 can remove objects near the equator of the image. To remove objects above or below the camera, such as the tripod, an adaptation must be made. Removing the tripod is achieved by rotating the panorama so the object appears near the equator, removing the object as in section 3, and then reversing the rotation to return the panorama to its original orientation. Rotating the panorama can be thought of as choosing different poles on the viewing sphere before projecting to equirectangular, as shown in figures 9a and 9b. However, some issues are introduced by this method of performing cuts.

As indicated in section 3.3, a circular distortion is introduced at the poles during the cut. If the poles during the cut are in fact



(a) Equirectangular following large cut.



(b) Equirectangular is non-homogeneously stretched in the x axis to restore 2:1 aspect ratio. Middle area is not stretched to preserve central figure.

Figure 8: Grid shows non-homogeneous stretching used to preserve salient content.

from the equator of the original panorama, these distortions become highly noticeable. As can be seen in figure 9, the horizon becomes bent. There are a number of ways to mitigate this effect. The first is to remove an equally sized cut from the opposite side of the sphere. This means the horizon stays in place. This can produce plausible results in some cases. However in some cases this will only increase the perceived distortion, particularly when there are straight lines such as buildings, such as in figure 9g.

An alternative approach is to non-homogeneously stretch the content as described in section 4.1, specifically, stretching only the content below the horizon to return the 2:1 aspect ratio following a cut. This allows the horizon to remain flat, and preserves the content above the equator at the expense of some additional stretching below the equator. An example of this can be seen in figure 9h.

It is important to note that this method of tripod removal requires two rotations of the sphere to be stored as equirectangular images – the first to move the pole to the equator, and the second to move it back after editing. Each of these rotations uses a filter that distorts content at the poles, resulting in reduced clarity of the image. A way to mitigate this is to work in a higher resolution than required for the final image before scaling down, although this will also increase computation time.

4.3 Variable FOV expansion

The algorithm described in section 3.1 can be improved by noting that a better cut may be found if we are willing to lose more of the original panorama. With the overlap size fixed, the optimisation is performed at increasing HFOVs within a range that is deemed allowable. Each increase in the HFOV is equivalent to having removed an additional column in the initial cut stage i.e. the width of section A in figure 1b is increased one column at a time. The best join is then identified by taking the cut with the minimum cost. This can produce better results in many situations, but it is



(a) Original equirectangular – tripod is stretched across the south pole. (b) The sphere is rotated so the tripod is at the equator.



(c) Before the cut: looking down, the tripod is visible. (d) After cut: tripod removed using tripod removal using FOV expansion and graph cuts.



(e) Before the cut: the horizon is straight. (f) After cut: circular distortion at the pole has warped the horizon.



(g) Straightening the horizon: taking an equal sized cut from the opposite side of the sphere. (h) Straightening the horizon: non-homogeneous stretching of content below the horizon preserves buildings better.

Frame courtesy of the BBC

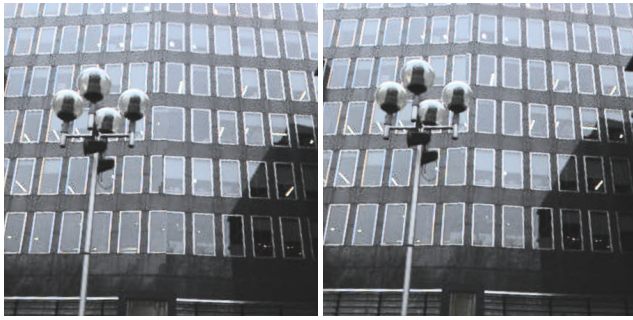
Figure 9: Tripod removal using FOV expansion and graph cuts.

most noticeable when a repeated pattern is visible and a sensible alignment would produce better results, as is the case in figure 10.

4.4 Weight adjustment

As the algorithm described in section 3.1 performs cuts on equirectangular images, the edge weights used during the optimisation are not completely correct. Due to the heavy distortion at the poles present in equirectangular images, a pixel at the pole of the viewing sphere takes up substantially more pixels in the equirectangular image than a pixel from the equator. Despite this, a pixel's latitude does not alter the assigned weight in the graph cuts formulation. We altered this by increasing the weights nearer the equator, adding a term inversely proportional of the pixel's size once projected onto the viewing sphere. This meant pixels near the equator would have a higher weight and contribute more to the optimisation.

While this weight formation may be technically more correct, we were unable to produce a result that was noticeably superior using this method. It was suspected that the generally narrow size of the overlapping region means that the cut chosen at the equator is influenced very little by the choice of cut at the poles.



(a) Overlap alignment leads to poor cut. (b) Better alignment has produced a better cut.

Figure 10: A better cut can be found by allowing a variable FOV expansion, particularly in areas with a repeated pattern.

5. INPAINTING

The method described in section 3 changes the FOV of the image, resulting in warping of the content. Additionally, to perform tripod removal, the poles must be moved twice, each using a filter that increases blurring. Instead, inpainting could be used to remove unwanted objects or fill holes. To perform inpainting, the object or hole to be removed is masked to identify the area to be inpainted. Content from the rest of the image is then used to fill this area in a way that appears plausible. In all of our examples this mask was created manually, and the inpainting performed using Adobe Photoshop’s content-aware fill [1].

5.1 Inpainting in equirectangular

The most simple method to inpaint objects in panoramic images is to inpaint directly in equirectangular. In many cases, this will produce excellent results and no further work is required. Examples of this can be seen in figure 11. However there are many situations in which this method will not work. Inpainting the tripod, for example, is infeasible using this method as the tripod is normally at the south pole – an area so distorted in an equirectangular projection that inpainting cannot produce plausible results. Indeed, even masking the tripod for inpainting is a challenge as the distortion is so strong that it is difficult to identify it definitely.

One method to combat these issues is to rotate the tripod to the centre of the equirectangular image, perform inpainting, and then rotate the panorama back to its original orientation. These rotations can be performed in a similar fashion to section 4.2. In this case, however, the effects of the filter used during rotation can be undone at the end of the process. This is achieved by rotating the mask created during the inpainting step through an identical rotation as that of the image. This mask is then used to copy the inpainted material from the final image into the original, unrotated image. To prevent noticeable discontinuities the copied section can be feathered in. As can be seen in figure 12, this method is capable of producing excellent results.

An issue with inpainting in an equirectangular image is that it can produce poor results in some situations. Content can be used for inpainting that does not match the distortion of the hole being filled. This is particularly noticeable when inpainting geometric textures. To highlight this, a challenging issue was considered. Some 360° cameras do not capture a full sphere. Point Grey’s Ladybug3, for example, does not have a downward facing camera. This results in a black hole covering the south pole. Removing this black hole follows a similar process as removing an unwanted object – a section of the media must be removed or replaced while maintaining



(a) Closeup of object to remove. (b) Inpainted in equirectangular. Frame courtesy of Peter Boyd Maclean



(c) Closeup of object to remove. (d) Inpainted in equirectangular. Frame courtesy of the BBC

Figure 11: Inpainting directly in equirectangular.

a plausible visual result. As can be seen in figure 13, inpainting this large hole with geometric content causes issues. In figure 13d, brickwork from elsewhere in the equirectangular image is used, resulting in an image that is noticeably incorrect.

5.2 Straight line preserving projections

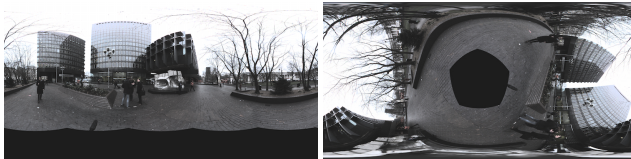
A possible solution to the problems seen in figure 13d is to inpaint in a projection that preserves straight lines. A cubic projection is one in which the sphere is projected onto the 6 sides of a cube. Each cube face is a rectilinear image with a HFOV and VFOV of 90°. While this projection has issues in terms of storage and display – being somewhat complicated to understand when viewed in flat form – it has the advantage of having little distortion on any of the cube’s faces. Importantly, straight lines appear straight in each tile. By inpainting using only the bottom tile of a cubic projection, the hole shown in figure 14a can be removed with fairly good results, as shown in figure 14b. However, using only the bottom tile of the cube means that less of the panorama is available for matches to be located. For many cases this will not be an issue as the best content to inpaint with will likely be near the hole. For complex inpainting tasks requiring more information, another approach may improve results.

The inpainting of geometrically complicated surfaces is an example of a case where improvements may be possible over a cubic projection. This can be seen in figure 14d, where the 130° VFOV rectilinear image centred on the hole shown in figure 14c is inpainted. At 130° VFOV, content at the edges of the rectilinear image undergo substantial stretching distortion. In the cubic version in figure 14b, the results are crisp but there is noticeable repetition. In figure 14d, repetitions are less obvious, although this comes at the expense of clarity as the results are more blurred. This makes sense – the rectilinear version in figure 14c has more choice as it covers a larger area, however stretching distortions at the edges of

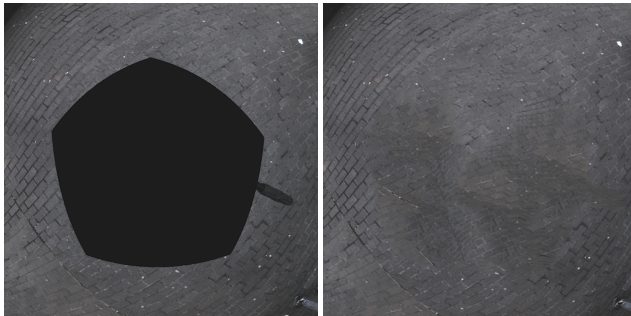


(a) Closeup of the tripod. (b) Tripod is inpainted.
Frame courtesy of the BBC

Figure 12: Inpainting in equirectangular following rotation of south pole to the equator, as in figures 9a and 9b.



(a) Original equirectangular. (b) Hole rotated to equator.



(c) Closeup of hole to be filled. (d) Inpainted in equirectangular.

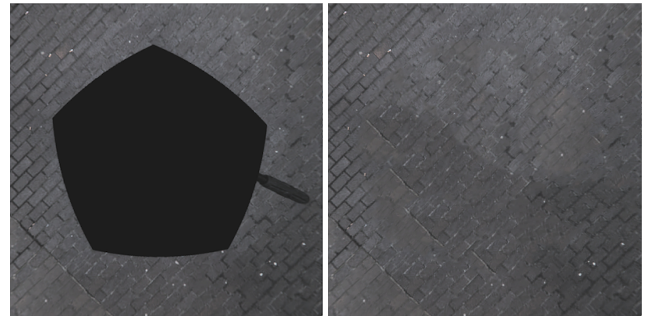
Figure 13: Failure case of inpainting in equirectangular: large hole at south pole inpainted with geometric texture.

the rectilinear mean blurred content is available for use by the inpainting algorithm.

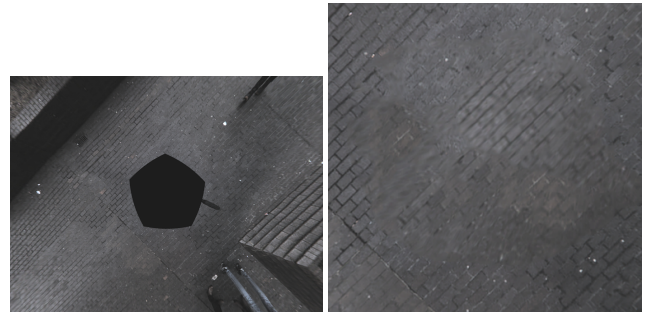
Whether or not a cubic projection is sufficient will depend on the size of the hole and the type of content being inpainted. Inpainting could be performed on rectilinear views with differing FOVs, allowing the best result to be found. Inpainting the 90° HFOV/VFOV rectilinear tile of the cubic projection has the advantage that the inpainted tile can be swapped in for the original tile and no more work is required. For rectilinear views with other FOVs, the inpainted rectilinear content must be stitched back into the original panorama, adding an additional step to the process. Working in a higher resolution than required for the final panorama may also be useful, as stretched content at the edges of the rectilinear will be better quality and therefore produce superior results if used by the inpainting algorithm.

5.3 Limitations

Inpainting panoramic content can produce good results in many cases. It suffers similar limitations, however, to inpainting standard format content. There are times when inpainting cannot produce a plausible result. Examples of this can be seen in figure 15. In figure



(a) Bottom cubic tile with hole. (b) Inpainted in cubic.



(c) 130° VFOV rectilinear with hole. (d) Closeup of inpainted rectilinear.

Figure 14: Inpainting of large hole at south pole.

15b, the implausible removal of a tree's trunk mean the resultant image is unconvincing. In figure 15d, revealed structure cannot be realistically constructed using content found elsewhere in the image.

In such cases, it may be better to remove the area entirely using the method described in section 3, and seen in figure 15e. In some cases, however, neither method will work, as shown in figure 15f where a camera operator cannot be removed while retaining a plausible image. The substantial change in background and the geometric structure of the surrounding room prevent the successful application of the graph cuts method.

The completion of the nadir using inpainting shown in figure 14 is very successful, even in the presence of complex geometric textures. However, this is in part due to the fact that the content is viewed fronto-parallel, as the camera's downward vector is perpendicular to the ground plane. To inpaint geometrically textured content on surfaces that are not viewed fronto-parallel, it may be beneficial to correct the perspective distortion prior to inpainting, as in the work of Pavić et al. [12].

6. VIDEO

Extending the technique described in section 5 to video is easy in certain situations. If the camera is fixed, the background is static, and foreground objects do not pass in front of the inpainted hole, the results of inpainting in one frame can be copied to the other frames. Small changes in global illumination can be handled by adjusting the brightness of the inpainted section to match the target frame before transfer.

While these constraints may seem restrictive, tripod removal for a static camera often fulfils the required criteria. This assumes other foreground objects or shadows do not cross the inpainted section. This method was used to remove the tripod from a video sequence, the results of which can be seen in figure 16.

The object removal technique described in section 3 could be ex-



(a) Closeup of large object to re-move. (b) Revealed tree trunk is not reconstructed by inpainting.



(c) Camera operator should be re-moved. (d) Inpainting does not realistically complete the door.

Frame courtesy of the BBC



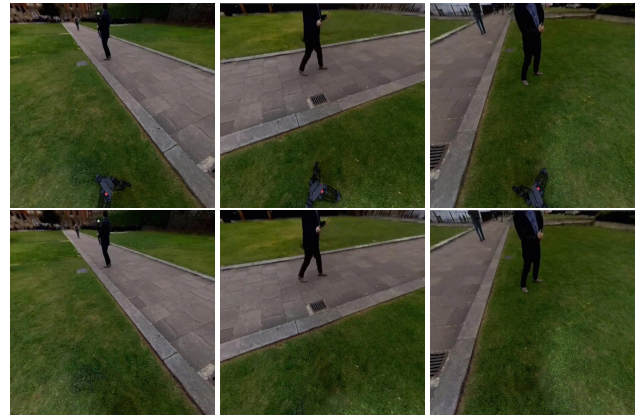
(e) Graph cuts applied to image 15a: the result is better than 15b. (f) Graph cut applied to image 15d: the result is worse than 15d.

Figure 15: Failure cases of inpainting method. Inpainting does not always produce plausible results when the removed object is very large or novel shapes are revealed. Our graph cuts method can improve results in some cases.

tended to video in a similar way, applying the cut found in the first frame to each successive frame in the video. This method would suffer from similar limitations as well – foreground objects could not move across the cut area, and dynamic backgrounds would not be supported.

7. CONCLUSION

It has been shown that two methods for object removal are viable for use in 360° panoramas. Inpainting can produce excellent results in many situations. In many cases where the object to remove is not at the poles, it was shown that the standard equirectangular projection can be used without additional steps. Due to the distortion characteristics of equirectangular images, however, more needs to be done to inpaint content at the poles. Tripod removal was shown to work well by adding rotation steps to the process, while cubic and rectilinear projections are required for complex inpainting tasks such as geometric textures. Inpainting failed to produce



Frame 1 Frame 250 Frame 500
Video courtesy of the BBC

Figure 16: 100° FOV rectilinear views of an inpainted video. The inpainted section was copied between the equirectangular frames. Above, the tripod can be seen at the bottom of the unedited frames. Below, the tripod has been successfully removed.

good results, however, in some cases where the background being revealed could not be plausibly reconstructed using content in the image.

In these cases, the FOV expansion and graph cuts method described in section 3 could be used. Instead of attempting to reconstruct the background revealed during object removal, the entire section is removed and the neighbouring content cut together in a plausible way. This method was shown to produce good results including when the object being removed was not at the equator. Methods were also discussed to alleviate the distortions this method can introduce, such as seam carving and non-homogeneous warping. However, in some situations such as the presence of very regular surrounding structure, this method may fail to produce acceptable results.

8. FUTURE WORK

The most obvious future direction for this research is further extensions to video. For scenes with static backgrounds in which objects do not move through the removed area, it was shown that inpainted material can be copied between frames. A similar process may work for the method described in section 3, by applying the same cut to each frame. For dynamic backgrounds and foreground objects that move across the removed section, however, more work must be done to maintain a realistic effect. For the graph cuts solution, this could involve performing the cut in a spatio-temporal volume as proposed in the Graphcut Textures paper [9]. For inpainting in video, one of the spatio-temporal extensions could be investigated, such as that of Newson et al. [11].

The method proposed in section 3 removes the object and then performs a cut on the overlapping area. This may remove more of the original panorama than necessary. In future we would like to investigate ways to mask the object to be removed and allow the algorithm to find a good overlap and cut in order to produce the best result with the minimum amount of loss.

An interesting area to explore would be the arbitrary placement of the poles when projecting to equirectangular, i.e. poles that are not necessarily opposite each other on the viewing sphere. While these points could no longer strictly be called “poles”, an invertible mapping in which arbitrary points on the viewing sphere were

stretched across the top and bottom of the equirectangular image would allow less of the original panorama to be removed during a cut. Other attributes could also be investigated, such as allowing the director to assign a larger amount of the image pixels to a salient area.

Other possible avenues of research include extending these techniques to stereoscopic panoramas. Additionally, it may be useful to build a tool to help crews on site identify where best to place equipment to facilitate easy removal in post-production.

9. ACKNOWLEDGMENTS

This work was supported in part by grant EP/G037159/1 for University College London's Virtual Environments, Imaging & Visualisation Doctoral Training Centre (<http://engdveiv.ucl.ac.uk/>) from the UK Engineering and Physical Sciences Research Council (EPSRC) and the BBC. The authors would like to thank BBC R&D for their support and allowing us to use their content.

10. REFERENCES

- [1] Adobe. Content-aware fill. <http://adobe.com/technology/projects/content-aware-fill.html>. Accessed: 2015-06-09.
- [2] S. Avidan and A. Shamir. Seam carving for content-aware image resizing. In *ACM Transactions on graphics (TOG)*, volume 26, page 10. ACM, 2007.
- [3] C. Barnes, E. Shechtman, A. Finkelstein, and D. Goldman. Patchmatch: A randomized correspondence algorithm for structural image editing. *ACM Transactions on Graphics-TOG*, 28(3):24, 2009.
- [4] M. Brown and D. G. Lowe. Automatic panoramic image stitching using invariant features. *International journal of computer vision*, 74(1):59–73, 2007.
- [5] R. Carroll, M. Agrawala, and A. Agarwala. Optimizing content-preserving projections for wide-angle images. *ACM Transactions on Graphics-TOG*, 28(3):43, 2009.
- [6] A. Criminisi, P. Pérez, and K. Toyama. Object removal by exemplar-based inpainting. In *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, volume 2, pages II–721. IEEE, 2003.
- [7] H. Igehy and L. Pereira. Image replacement through texture synthesis. In *Image Processing, 1997. Proceedings., International Conference on*, volume 3, pages 186–189. IEEE, 1997.
- [8] R. Keys. Cubic convolution interpolation for digital image processing. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 29(6):1153–1160, 1981.
- [9] V. Kwatra, A. Schödl, I. Essa, G. Turk, and A. Bobick. Graphcut textures: image and video synthesis using graph cuts. In *ACM Transactions on Graphics (ToG)*, volume 22, pages 277–286. ACM, 2003.
- [10] J. Lowensohn. YouTube now supports 360-degree videos. <http://www.theverge.com/2015/3/13/8203173/youtube-now-supports-360-degree-videos>. Accessed: 2015-06-17.
- [11] A. Newson, A. Almansa, M. Fradet, Y. Gousseau, and P. Pérez. Towards fast, generic video inpainting. In *Proceedings of the 10th European Conference on Visual Media Production*, page 7. ACM, 2013.
- [12] D. Pavić, V. Schönefeld, and L. Kobbelt. Interactive image completion with perspective correction. *The Visual Computer*, 22(9-11):671–681, 2006.
- [13] A. Pratley. Which 360 video camera should I buy? <http://www.beamify.tv/blog/which-360-video-camera-should-i-buy/>. Accessed: 2015-06-27.
- [14] Y. Pritch, E. Kav-Venaki, and S. Peleg. Shift-map image editing. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 151–158. IEEE, 2009.
- [15] M. Rubinstein, A. Shamir, and S. Avidan. Improved seam carving for video retargeting. In *ACM transactions on graphics (TOG)*, volume 27, page 16. ACM, 2008.
- [16] L. K. Sacht, P. C. Carvalho, L. Velho, and M. Gattass. Face and straight line detection in equirectangular images. In *Workshop de Visão Computacional. Presidente Prudente, SP, Brasil: FTC-UNESP*, pages 101–106, 2010.
- [17] H. S. Sawhney and R. Kumar. True multi-image alignment and its application to mosaicing and lens distortion correction. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 21(3):235–243, 1999.
- [18] J. P. Snyder. *Map projections—A working manual*. Number 1395. USGPO, 1987.
- [19] J. P. Snyder. *Flattening the earth: two thousand years of map projections*. University of Chicago Press, 1997.
- [20] R. Szeliski and S. B. Kang. Direct methods for visual scene reconstruction. In *Representation of Visual Scenes, 1995. (In Conjunction with ICCV'95), Proceedings IEEE Workshop on*, pages 26–33. IEEE, 1995.
- [21] R. Szeliski and H.-Y. Shum. Creating full view panoramic image mosaics and environment maps. In *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, pages 251–258. ACM Press/Addison-Wesley Publishing Co., 1997.
- [22] Y. Wexler, E. Shechtman, and M. Irani. Space-time completion of video. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(3):463–476, 2007.
- [23] L. Wolf, M. Guttman, and D. Cohen-Or. Non-homogeneous content-driven video-retargeting. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–6. IEEE, 2007.