Eurographics Symposium on Virtual Environments (2006)
Roger Hubbold and Ming Lin (Editors)

GA based Adaptive Sampling for Image-based Walkthrough

Dong Hoon Lee[1], Jong Ryul Kim †1 , and Soon Ki Jung[2]

[1]Div. of Digital Contents, Dongseo University, S. Korea
[2]Dept. of Computer Engineering, Kyungpook National University, S. Korea
[1]{dhl, xmaskjr}@dongseo.ac.kr, [2]skjung@knu.ac.kr

Abstract

This paper presents an adaptive sampling method for image-based walkthrough. Our goal is to select minimal sets from the initially dense sampled data set, while guaranteeing a visual correct view from any position in any direction in walkthrough space. For this purpose we formulate the covered region for sampling criteria and then regard the sampling problem as a set covering problem. We estimate the optimal set using Genetic algorithm, and show the efficiency of the proposed method with several experiments.

Categories and Subject Descriptors (according to ACM CCS): I.3.7 [Computer Graphics]: Virtual Reality

1. Introduction

Image-based rendering (IBR) generates novel views from a set of input images instead of 3D models. Among the many IBR approaches, one promising IBR approach enhances the images with per pixel depth. This allows warping the samples from the reference image to the desired image. Generally the image-based approach using depth is called as image-based rendering by warping (IBRW) [MB95]. However, in IBRW, simply warping the samples does not guarantee high-quality results because one must reconstruct the final image from the warped samples. To solve this problem, most of all, good reconstruction algorithms such as efficient warping, splatting etc. are needed [MMB97, McM97]. But the fundamental and important problem of properly sampling has remained largely unanswered.

The sampling is a very difficult problem since the sampling rate will be determined by the scene geometry, the texture on the scene surface, the reflection property of the scene surface, the specific IBR representation we take, the capturing and the rendering camera's resolution, etc [Zha04]. Over-sampling was widely adopted in the early stages [LH96, SH99]. Generally, to reduce the huge amount of data due to over-sampling, many compression techniques are utilized [Zha04] instead of dealing with the sampling problem. However, sampling is more of a fundamental problem to IBR.

In this paper, we basically deal with sampling problem. Our goal is to select minimal sets from the initially dense sampled data set, while guaranteeing a visually correct view from any position in any direction. For this purpose we regard the sampling problem as a set covering problem and estimate the optimal set using Genetic Algorithm.

The rest of the paper is organized in the following manner. In Section 2, we provide previous related works, and in Section 3 and 4 define the coverage error threshold and deduce the set covering problem from the covered region, respectively. In Section 5, a GA-based method for optimizing critical views is proposed. We propose an integration method of the optimized set to guarantee the scalability in Section 6 and present the efficiency of the proposed method with several experiments in Section 7. Finally we conclude and present future work.

2. Background and Previous Work

Depending on how the capturing cameras are placed, IBR sampling can be classified into two categories: uniform sam-

† Program for the Training of Graduate Students for Regional Innovation: This research was supported by the Program for theTraining of Graduate Students in Regional Innovation which was conducted by the Ministry of Commerce Industry and Energy of the Korean Government

pling and non-uniform sampling. In uniform sampling, the cameras are positioned evenly on a capture configuration which is usually a surface or a line. The light field [LH96] and the concentric mosaics [SH99] are the representative examples. In the case of uniform sampling, the main research topic is to find the minimum sampling rate or largest spacing between cameras such that one can achieve perfect reconstruction quality on the navigation space. The goal of non-uniform sampling analysis is also to find the minimum number of cameras while rendering the highest quality scene. But in this case, arranging camera position is another important problem.

In practice, objects in the scene have varying surface properties. For instance, if a scene has non-Lambertian surface or occluded regions, more samples are needed. In general, a real world scene is composed of Lambertian and non-Lambertian surfaces. The Lambertian surface may need a low sampling rate, while the non-Lambertian surface may need a high sampling rate. Uniformly sampling the scene without concerning about the regional surface property may easily cause over-sampling of the Lambertian surface or under-sampling of the non-Lambertian surface [Zha04]. It is therefore natural to consider non-uniform sampling. One thing to notice is, in uniform sampling, since the sampling is periodic, we only need to tell how many images/light rays are needed for perfect reconstruction of the scene; in non-uniform sampling, however, we need to answer not only how many images are needed but also where to place these cameras. In this paper, the non-uniform sampling approach is adopted.

Fleishman et al. [FCOL00] proposed an automatic camera placement algorithm for IBR. They assumed a mesh model of the scene is known. The goal is to place the cameras optimally so that the captured images can form the best texture map for the mesh model. They proposed an approximation solution for the problem by testing a large set of camera positions and selecting the ones with higher gain rank. Here the gain was defined based on the portion of the image that can be used for the texture map. However, this method cannot be extended to the sampling of real environment due to the assumption of the known mesh model and is only applicable to scenes with Lambertian surfaces.

Schirmacher et al. [SHS99] proposed an adaptive acquisition scheme for a LightField setup. Assuming the scene geometry is known, they added cameras recursively by predicting the potential improvement in rendering quality when adding a certain view. They asserted a-priori error estimator accounts for both visibility problems and illumination effects such as specular highlights used to some extent. However, this error estimator definitely has its limits unless a ground truth data is obtained.

Depending on the application scenario, solutions to the non-uniform sampling problem can be classified into two categories: incremental sampling and decremental sampling [Zha04]. In incremental sampling, the samples are captured one by one incrementally. The stopping criterion is either the overall number of samples desired reached, or an error requirement that the sampling process must achieve. In decremental sampling, we assume there is already a dense set of samples available that fulfills the error requirement. This sample set might be too large, thus decremental sampling can be used to reduce the size, while keeping the error within the requirement.

Most previous image-based walkthrough systems which have the sampling strategy have adopted the incremental sampling approach since it was too difficult to get the initial dense samples. However, recent advances in acquisition and modeling technologies have resulted in large databases of real-world and synthetic environments. This is made possible by using an omni-directional camera capture system in which the camera is placed on a motorized cart together with a battery, computer, frame grabber and fast disk system to store the captured images on [AFC02]. Therefore, in this paper, we choose the decremental sampling approach to determine optimal view positions.

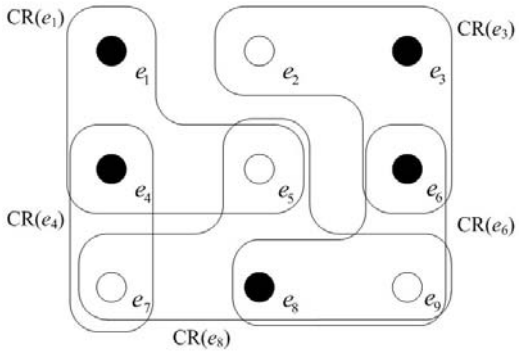

Figure 1: *An example of the sampling configuration.*

3. The Problem Statement

3.1. The Covered Region

The sampling method in the decremental approach can be expressed as a view selection problem and the optimized views imply a kind of best view which is representative of the sampling space. As reconstruction algorithm, we use a 3D warping algorithm [MB95] for predicting the view point location. We estimate reconstruction quality when sampled image at each view point location is warped into other view point locations. The computation of the reconstruction quality is based on estimating the warping error including holes and color mismatching error. Potential positions with a small error, when a sampled position (source) warps into other positions (destinations), are considered as elements of the covered region at a source position. Figure 1 shows an example of sampling configuration, where CR(e_n) a set of elements

which is covered by source image e_n. In addition, the filled circles show the sampled view position and the blanked circles indicate the non-sampled view position.

Our goal is to minimize the number of the sampled view position while all elements must be covered by the selected view position, which means the selected view positions guarantee adequate reconstruction qualities at all positions for navigation. In other words, the elements covered by a single set have a strong resemblance between them. This problem can be modeled as set covering problem, which is one of the typical problems in combinatorial optimization

3.2. Set Covering Problem

The set covering problem (SCP) is the problem of covering the rows of an m-row, n-column, zero-one matrix (a_{ij}) by a subset of the columns at minimal cost. Considering a vector $\mathbf{x}$, such that $x_j = 1$ if column j (with a cost $c_j > 0$) is in the solution and $x_j = 0$ otherwise. The set covering problem is then formulated as:

$$\text{Minimize} \quad \sum_{j=1}^{n} c_j x_j \tag{1}$$

$$\text{Subject to} \quad \sum_{j=1}^{n} a_{ij} x_j \geq 1, \quad i = 1,\ldots,m \tag{2}$$

$$x_j \in \{0,1\}, \quad j = 1,\ldots,n. \tag{3}$$

Equation 2 ensures that each row is covered by at least one column and equation 3 is the integrality constraint. If all the cost coefficients are equal, the problem is called the unicost SCP. The SCP has been proven to be NP-complete [GJ79].

In our problem, each row indicates view position on navigation space and each column means the sampled view position. The cost c_j can be formulated by the error values obtained from each element of the selected set. The sampling configuration in Figure 1, therefore, can be formulated as the set covering problem with 9 rows and 5 columns under an assumption of unicost.

$$A^T = \begin{bmatrix} 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}.$$

A feasible solution to the instance is $\mathbf{x} = [1,\ 1,\ 0,\ 1,\ 0]$ with cost 3.

4. Estimating the Covered Region

From the initial dense samples, the covered region can be obtained at the each view point, which is performed by estimating the reconstruction error. The computation of the reconstruction error is based on warping the target image to the candidate point and estimating the dissimilarity between the warped image and the sampled image at the candidate point. As mention before, in order to estimate the error associated with any candidate viewpoint, we have chosen image-based rendering by warping (IBRW) as the basis for the reconstruction of the candidate view point. McMillan and Bishop [MB95] show how to compute the desired image coordinates of a depth image sample using the 3D warping equations.

$$u_2 = \frac{w_{11} + w_{12}\cdot u_1 + w_{13}\cdot v_1 + w_{14}\cdot \delta(u_1,v_1)}{w_{31} + w_{32}\cdot u_1 + w_{33}\cdot v_1 + w_{34}\cdot \delta(u_1,v_1)}, \tag{4}$$

$$v_2 = \frac{w_{21} + w_{22}\cdot u_1 + w_{23}\cdot v_1 + w_{24}\cdot \delta(u_1,v_1)}{w_{31} + w_{32}\cdot u_1 + w_{33}\cdot v_1 + w_{34}\cdot \delta(u_1,v_1)},$$

where u_2, v_2 are the desired image coordinates, u_1, v_1 the original (reference) image coordinates, the w's are transformation constants obtained from the reference and desired image camera parameters, and $\delta(u_1,v_1)$ is the generalized disparity at sample u_1, v_1, which is defined as the ratio between the distance to the reference image plane and $z_{eye}(u_1,v_1)$.

Reconstructing by simply setting a desired image pixel to the color of the sample that warps within its boundary results in holes thus is not acceptable. Also, more than one visible sample can warp to the same pixel, and simply discarding all but one sample produces aliasing. To overcome these problems, we use the Popescu's Forward Rasterization algorithm [Pop01]. Throughout the each per-pixel error, the total error for the whole view can be estimated by the sum of square error.

Figure 2 shows the process of error estimation. Figure 2(a) and (b) indicates the reference and the desired image, respectively. By 3D image warping, we obtain the warped image at the candidate view point as shown in Figure 2(c). By applying the Forward Rasterization algorithm, visibility holes are removed and this shows a reasonable visual quality in Figure 2(d). The dissimilarity between the warped image (Figure 2(d)) and the sampled image (Figure 2(b)) at the destination point determines whether the reference image covers the destination position.

Since we already sampled all images at every sample position, the evaluation of the performance is achieved by ground truth data using the following two quality measures:

1. RMS (root-mean-squared) error between the computed warped image $d_j^{(B)}$ and the sampled ground truth image $d_j^{(T)}$.

$$R = \left(\frac{1}{N} \sum_{j=0}^{N-1} \left| d_j^{(B)} - d_j^{(T)} \right|^2 \right)^{\frac{1}{2}}, \tag{5}$$

where N is the total number of pixels in the destination image.

2. Percentage of bad matching pixels,

$$B = \frac{1}{N}\sum_{j=0}^{N-1}\left(\left|d_j^{(B)} - d_j^{(T)}\right| > \delta_d\right), \qquad (6)$$

where δ_d is a disparity error tolerance.

Accordingly, the total reconstruction error can be expressed as follows:

$$E = \alpha R + \beta B. \qquad (7)$$

We select the weight of RMS (α) as 10.0 and the weight of percentage of bad matching pixels (β) is 0.1. We think the bad matching pixels have a great influence on the quality of scene and adjusted the parameters until we found an adequate weighting value. In addition, the disparity error tolerance δ_d sets to 15.0.

At this stage, for each position in the scene, the optimal rate is required to find the covered region. Such an approach would result in a very large number of reference images. Our approach, instead, is to compromise: rather than insisting that each pixel in the scene is covered with optimal quality, we are willing to allow lower coverage error threshold so long as the quality does not exceed some user-specific minimum Q. When the quality exceeds the coverage error threshold Q, the big error value is assigned. Eventually our goal is to select a small set of cameras such that all sets of elements are covered with quality smaller than Q.

Figure 3 shows the warped images according to the error value. The coverage error threshold can be adjusted according to the user's intension. In this paper, we selected the value as 1.85 by several experiments.

Figure 4 visualizes the error distribution of some candidate viewpoints. The sampling space consists of 450 images which are 30 images with respect to 15 paths. This figure shows an instance of the error distribution with respect to four corner points and one center point. We can find that each position has a different covered region according to the complexity of the local geometry.

5. Genetic Algorithm for SCP

The set covering problem has been proven to be NP-complete [GJ79]. Both optimal and heuristic solutions to this problem have been reported in the literature. Among many approaches for set covering problem, a genetic algorithm (GA) can be known as an 'intelligent' probabilistic search algorithm which can be applied to a variety of combinatorial optimization problems. Especially, the GA-based heuristic is able to generate optimal solutions for small-size set covering problems as well as to generate high-quality solutions for large-size set covering problems. In this paper, therefore, we solve the set covering problem with GA.

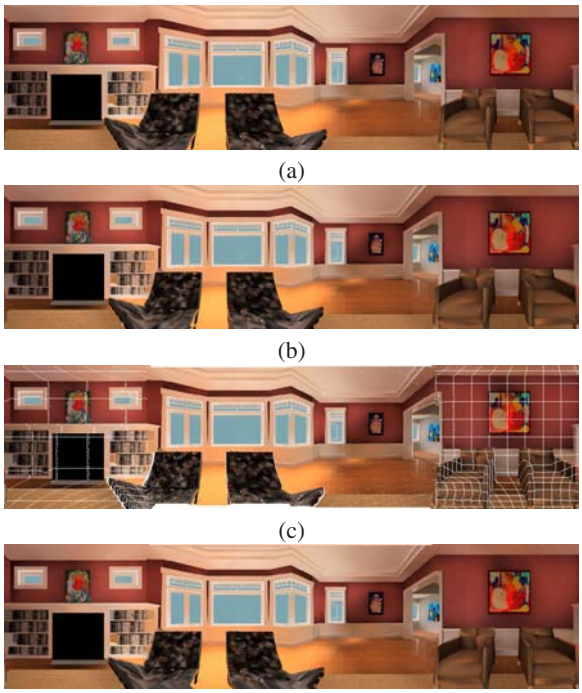

Figure 2: *An example of the 3D warping, (a) the reference image, (b) the desired image, (c) the warped image without hole filling and (d) the warped image generated by applying the Forward Rastrization algorithm.*

5.1. Representation and fitness function

The first step in designing a genetic algorithm for a particular problem is to devise a suitable representation scheme. Two representation schemes have been proposed for the set covering problem: column-based and row-based representation.

The column-based representation is an obvious choice for the SCP since it represents the underlying $0-1$ integer variables, i.e. for an n columns problem, an n-bit binary string can be used as the chromosome structure. A value of 1 for the i-th bit implies that column i is in the solution. The column-based representation of an individual's chromosome (solution) for the SCP is illustrated in Figure 5.

The initial population can be generated randomly. Note that individuals in the column-based representation are not guaranteed to the feasible. It means that not all rows will be covered. The problem of maintaining feasibility (all rows are covered) may be resolved by using a row-based representation. One possible representation is to have the chromosome size equal to the number of rows in the SCP. In this representation, the location of each gene corresponds to a row in the SCP and the encoded value of each gene is a column that covers that row (see Figure 6). With this repre-

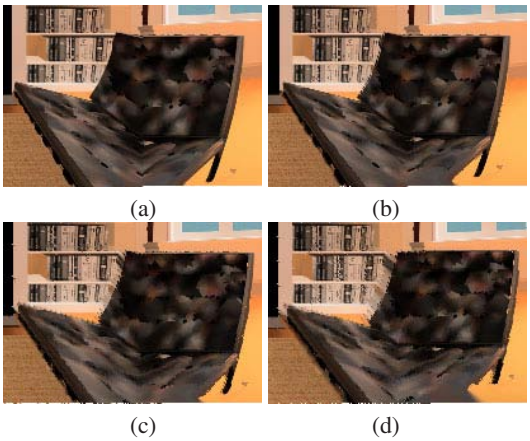

Figure 3: *The error images accroding to the quality measurment. The reference image is located at* 10*th image at path* 1*. (a)* 1*st image at path* 1 *(*$E = 1.103154$*), (b)* 1*st image at path* 2 *(*$E = 1.816137$*), (c)* 4*th image at path* 2 *(*$E = 1.9771854$*) and (d)* 5*th image at path* 4 *(*$E = 2.134124$*). This figure shows that defined low bound of quality is reasonable.*

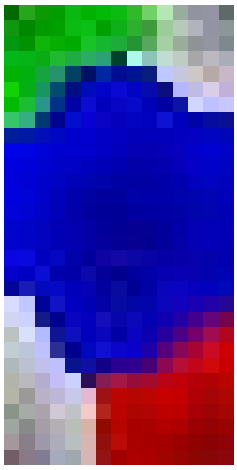

Figure 4: *Error distribution of a candidate view position. The sampling space consists of* 450 *images which are* 30 *images with respect to* 15 *paths. This figure shows an instance of the error distribution with respect to four corner points and one center point (dark = small errors, light = big errors).*

sentation, feasibility can generally be maintained throughout the crossover and mutation procedures. But the evaluation of the fitness may become ambiguous because the same solution can be represented in different forms and each form may give a different fitness depending on how the string is represented. Since the same column may be represented in more than one gene location, a modified decoding method for fitness evaluation is used by extracting only the unique

column(gene)	1	2	3	4	5	…	*n*-1	*n*
bit string	1	0	1	1	0	…	1	0

Figure 5: *Column-based representation.*

set of columns which the chromosome represents (i.e. repeated columns are only counted once). This paper, therefore, adopts the row-based representation.

row(gene)	1	2	3	4	5	…	*m*-1	*m*
string	10	7	10	21	5	…	49	7

Figure 6: *Row-based representation.*

In the row-based representation, each gene has a cost which is an error rate of the selected set with respect to the row. Throughout the optimization with sum of cost, we can guarantee the selected sets have optimal image quality while the number of the set is minimized. In other words, each element, which is covered by several sets, selects the set with minimum cost although the number of set is already optimized.

5.2. Genetic Operators

Selection: The tournament selection is one of many methods of selection in GAs which runs a tournament among a few individuals and selects the winner (the one with the best fitness). Selection pressure can be easily adjusted by changing the tournament size. If the tournament size is higher, weak individuals have a smaller chance of being selected. The tournament selection has several benefits: it is efficient to code, works on parallel architectures and allows the selection pressure to be easily adjusted. The selection process consists of two stages:

1. Select a group of N ($N \geq 2$) individuals, which N is tournament size.
2. Select the individual with the highest fitness from the group.

Crossover: Crossover is implemented with uniform crossover operator (or called multi-point crossover) given by Syswerda [Sys89], which has been shown to be superior to traditional crossover strategies for combinatorial problem. Uniform crossover first generates a random crossover mask and then exchanges relative genes between parents according to the mask. A crossover mask is simply a binary string with the same size of chromosome. The parity of each bit in the mask determines, for each corresponding bit in an offspring, from which parent it will receive that bit form.

Mutation: Mutation is performed as random perturbation within the permissive range from 1 to n (represents the number of sets which a row belongs to).

6. Integration: Sub-region Zippering

The row-based GA representation has the chromosome size equal to the number of elements in the scene which we want to walkthrough. Since our captured data sets usually contain thousands of images, the optimization of the whole scene's elements generates too large a chromosome size. This approach is unrealistic in situations such as ours where the size of the GA representation exceeds the capacity of host memory.

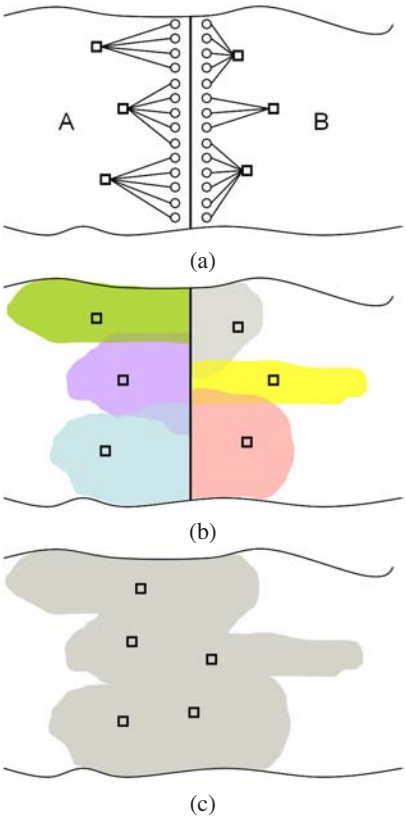

Figure 7: *The process of sub-region zippering. The blanked box indicates the selected set and the blanked circle means each element. (a) The critical view selection algorithm is performed at the sub-regions A and B, respectively and then selects sets including the border elements. (b) The region which is covered by the selected sets is estimated. (c) With respect to the all elements and sets at the boundary region, we re-compute the optimal set.*

The first and most obvious option for overcoming the problem is to partition the whole scene and estimate the optimal solutions in the separate regions independently. This approach is very straightforward, but does not guarantee the optimal solution at the connected portions of each separate region. Fortunately, a little variance of a selected point does not affect the selection of other positions far away from the selected point. In other words, the effect of some variance of the selected point does not propagate global region. Therefore, the central step in combining the separated regions is the reorganization of sets which are located at the connected regions. The proposed optimal zippering strategy consists of three steps.

1. Select sets including the border elements.
2. Bound regions which consist of elements covered by the selected sets.
3. Reorganize the optimal sets from the selected bounded regions by the proposed GA mechanism.

7. Experimental results

7.1. Image acquisition

The system was implemented on a Pentium IV PC with a 1.7GHz CPU and 512Mbyte memory. We have captured a synthetic environment using 3D Studio Max. We captured three views at a location with 90° FOV and acquired a dense Sea of Images through a large environment (see Figure 8). Four color and depth map images at the same camera position were stitched into a panoramic image. At that time, we captured the rays with regular angular resolution. The amount of the test data is 450 images which are 15 stitched images with respect to 30 paths. The sample data is Figure 9 shows one of the synthesized panoramic images.

7.2. Computational results

In our computational study, 20 trials of the GA heuristic (each with a different random seed) were made for the test problem. Each trial terminated when the maximum generation is up to $20,000$. The population size was set to 500 for the problem and the coefficient of the mutation was 0.9 and the coefficient of crossover was 0.7. The best solution value diverges into 66 sets and the mean error of the set is 1.680670. This shows the proposed method generate sampling data with acceptable image quality.

To compare the performance of the proposed method, we implemented the k-means clustering algorithm. The number of cluster centers (k) was set to the number of the optimized set obtained from the proposed GA method. The position of the initial cluster centers were also chosen from the result of the proposed methods. The distance measure for clustering indicates the dissimilarity between two samples. Therefore we utilize the error measure is used for the distance as described in section 4. After classifying each sample according to nearest cluster center, each mean has to be recomputed using the re-classified samples. Generally the mean is updated by the sum of the square error. In our case, since the geometric mean does not imply a representative value among the samples in the same cluster, we select the mean as the element with the minimum error distance with respect to the all samples in the same cluster.

Figure 8: *A scene environment which we utilize for experiments. We captured* 15 *stitched images with respect to* 30 *paths.*

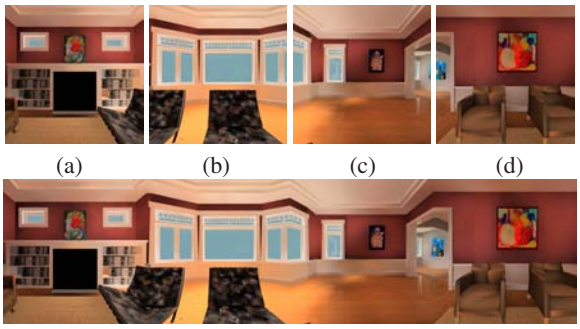

Figure 9: *Sample image of input panorama sequence. We have captured four views at the same location as illustrated in (a), (b), (c) and (d) respectively. (e) is a result of stitching with them.*

As the experimental result, the error value was obtained by 1.667272. This result shows the better solution although this is almost similar to the result of the first experiment. Since the *k*-means clustering procedure is well known as a form of stochastic hill climbing, which lead to local optima, this experimental result implies a possibility of the clustering procedure as a post refinement processing.

As another experiment, we compared the proposed method with the uniform sampling method (when given the same number of samples). For the uniform sampling, the multiple path-based capture configuration [DS05] is utilized and we refine the data with *k*-means clustering. The experimental result shows 2.139260 error values, which is inferior to the result of the proposed method.

Table 1 summarizes the experimental results of GA.

Table 1: *The experimental results of genetic algorithm with* 450 *sample data. The optimized critical set is diverged as* 66 *and the experiments (b) and (c) are achieved with the same number of samples.*

CVS (a)	*k*-means clustering after CVS (b)	Uniform sampling (c)
1.680670	1.667272	2.139260

7.3. Walkthrough on optimized data set

This section shows images of a test scene rendered with the method described. The test scene is captured with reference depth images according to the multiple path based capture configuration. The scene is synthetic and contains 48,000 primitives. This is a relatively simple model, but the performance of walkthrough is the same without the scene complexity. Figure 10 shows a sequence of images rotating from one point and Figure 11 shows novel views generated by translating along the arbitrary direction.

Figure 10: *Sequence of images rotating from one point.*

8. Conclusion and discussion

In this paper, we have presented an adaptive sampling method for image-based walkthrough. We first defined the covered region that is particularly suitable for the decremental sampling method and proposed an optimization approach, which is achieved by formulating the given sampling problem as set covering problem. We estimate the optimal set using Genetic algorithm, and also proposed the sub-region zippering method since global optimization of the whole scene elements generates too large chromosome size in GA. Finally we showed the efficiency of the proposed method with several experiments. The proposed sampling method provided simple and easy optimization process in large-scale scene walkthrough.

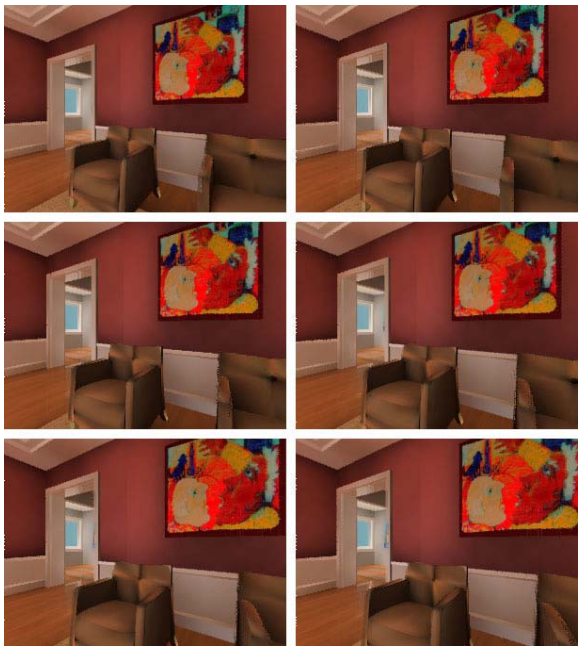

Figure 11: *Sequence of images by translating along the arbitrary direction.*

As part of future work, we plan to experiment the proposed method in real environments and try to enhance the proposed method with regard to the multiple reference images.

References

[AFC02] ALIAGA D., FUNKHOUSER T., CARLBOM I.: Sea of images. In *Proc. IEEE Visualization* (2002), pp. 331–338.

[DS05] D. LEE, S. JUNG: Multiple path-based approach to image-based street walkthrough. *Compute Animation and Virtual Worlds*

[FCOL00] FLEISHMAN S., COHEN-OR D., LISCHINSKI D.: Automatic camera placement for image-based modeling. *Computer Graphics Forum* (2000), 101–110.

[GJ79] GAREY M. R., JOHNSON D. S.: *Computers and Intractability: A Guide to Theory of NP-Completeness*. W.H. Freeman and Co., 1979.

[LH96] LEVOY M., HANRAHAN P.: Light field rendering. In *Proc. SIGGRAPH '96* (1996), pp. 31–42.

[MB95] MCMILLAN L., BISHOP G.: Plenoptic modeling: An image-based rendering system. In *Proc. SIGGRAPH '95* (1995), pp. 32–43.

[McM97] MCMILLAN L.: *An image-based approach to three-dimensional computer graphics*. Ph.D thesis UNC, 1997.

[MMB97] MARK W., MCMILLAN L., BISHOP G.: Post-rendering 3d warping. In *Symposium on Interactive 3D Graphics* (1997), pp. 7–16.

[Pop01] POPESCU V.: *Forward Rasterization: A reconstruction algorithm for image-based rendering*. Ph.D thesis UNC, 2001.

[SH99] SHUM H., HE L.: Rendering with concentric mosaics. In *Proc. SIGGRAPH '99* (1999), pp. 299–306.

[SHS99] SCHIRMACHER H., HEIDRICH W., SEIDEL H.: Adaptive acquisition of lumigraphs from synthetic scenes. *Computer Graphics Forum* (1999), 151–160.

[Sys89] SYSWERDA G.: Uniform crossover in genetic algorithms. In *Proc. 3rd Int. Conf. Genetic Algorithms* (1989), pp. 2–9.

[Zha04] ZHANG C.: *On sampling of image-based rendering data*. Ph.D thesis CMU, 2004.