First of all, we would like to thank all anonymous reviewers for their constructive and valuable comments, which will certainly help us improve our manuscript. We have revised the paper (changes colored in yellow). Here we provide response to the individual review comments.

**Summary Review**

1. A quantitative evaluation of the accuracy and a comparison with the alternative slower k nearest neighbor graph methods.

**Response:** We have added the matching error into Table 1; See the “”

Figure x (in …)shows the representative retrieval parts generated with the alternative slower kNNG methods.

1. A comparison to simply segmenting the database.

Response: The comparison to simply segmenting the database is shown in …

**Review 1**

1 The main question I have is if this method has any advantages over the existing kNNG methods other than query runtime performance. Table 1 clearly shows the queries are faster, at a cost of slower precompute. But I find it a bit vexing that there is no similar comparison of matching error, such as in Table 3. Reading between the lines, I am left with the impression that the contribution is only faster queries? This could still be important, but for most of the proposed applications I am skeptical.

**Response:** We have compared the matching error among different retrieval methods; See the last paragraph of “RC-kNNG Retrieval Performance” of Section 8.

1. why the descriptor as described in sec 5.1 is invariant to the index ordering of the contour polylines. IE, for two contours, the descriptor only makes sense if the (i,j) indices of the matrix refer to points in comparable relative spatial locations. In other words, where does 'i' start on each polyline?

**Response:** The contours are sampled in the counter-clockwise direction. In each section, ‘i’is the first sampling point in the sense of counter-clockwise direction.

3. Fig 12 shows a global symmetry example. Would local symmetry/similarity matching also be possible?

**Response:** Local symmetry/similarity matching are not possible. If the local symmetry have no obvious feature, our method could return a large number of results.

**Review 2**

1. Explanation of Recommendation: I would argue that the main novelty comes from putting together the complete system. The system contains multiple components that have been designed / adopted by the authors. The technical contribution of the individual parts is reasonable engineering, but I am not convinced that they can be claimed as major contributions.

**Response:** We have proposed novel algorithms for the sketch-based part retrieval problem: 1) the fast, sketch-based, partial 3D shape matching method based on multi-view projections; 2) the novel customized segmentation method based on a super-face graph.

2 One problem is the quality of the results. The system is demonstrated by some figures. However, in these figures, the retrieved results are not very similar to the sketches. This can be good or bad. In the good case, no great matches are available in the system and the system does a good job in retrieving the best available matches. In this case that indicates robustness. In the bad case, there would be many similar or better retrieval candidates that the system fails to identify. How can I tell? Therefore, the traditional evaluation of retrieval problems uses quantitative metrics, e.g. precision, recall, F-score, precision-recall curves, average precision, ... and a comparison of these metrics to competing algorithms. This submission does not provide quantitative results of the quality of the retrieval. Instead of analyzing the quality, the paper analyses the speed of the retrieval. The speed however is not relevant, without a clear demonstration of quality.

**Response:** We have made comparisons among different retrieval methods; See the last paragraph of the “RC-kNNG Retrieval Performance” part of Section 8.

2 The partial shape matching problem is not well specified. It is difficult to define a metric that computes the quality of partial matches and it is already difficult to get humans to agree on what partial matches are good or not. This is a general problem, also for other methods in this area, but I feel it could have been tackled a bit better.

**Response:** I agree with you that it is a difficult problem to propose a recognized metric that computes the quality of partial matches. We leave this for future (See the future work part …).

3 More retrieval results could be presented. Instead of showing more retrieval results, the paper contains very rough sketches of quite adventerous applications. I understand that other papers follow the same template, but the underlaying strategy is to replace quality with quantity.

**Response:** We have added two retrieval results in line (See the last scentence of the first paragraph of Section 8).

**Review 3**

1 I would not say that sketch-based modeling "typically" interprets sketches in a data-driven manner by searching a repository for a matching high-quality part. That is sketch-based shape retrieval, and only one, atypical approach to sketch-based modeling.

**Response:** We have reworded the sentence; See the first paragraph of Section 1.

1. Are contours sampled uniformly from directions around the unit sphere (Section 4, third paragraph)?

**Response:** Yes. The contours are sampled uniformly in the anticlockwise direction.

3 When sampling uncovered points on the contour ("Construction of the RC-kNNG" point 1), what happens when a point is sampled that does not have 10 consecutive points on either side? (Relatedly, 20 is an even number; how can a point be at the center of 20 consecutive points.)

**Response:** 1) Since the contour is end-to-end and the number of the sampled points in each contour is greater than 21, this condition nearly does not happen.

2) 20 is a mistake. We have corrected it to 21 (See "Construction of the RC-kNNG" point 1, “Contour Descriptor”Paragraph 3.)

4 Why must every one of the k-nearest neighbors be within a distance threshold for a section to be considered valid (last paragraph of Section 4)? Why not simply disconnect edges above the threshold? Does clustering the list mean clustering all valid sections? Is this clustering based on the same distance defined in Section 5.1?

**Response:** 1) If every one of the kNNs of the section be within a distance threshold, this section could be similar to others and have the potential to be as the seed section to represent other sections.

2) The edge is used for traversing (Section 5.2). We keep the edges above the threshold to establish more connectivity to the sections in the graph.

3) Yes. Clustering the list means clustering all the valid sections. We have reworded the scentence in the last paragraph of section 4.

4) Yes. The clustering is based on the same distance defined in Section 5.1. We have added one scentence in the last paragraph of Section 4.

5 How are super faces selected when the retrieved shape has multiple contours matching the query? Is the process described in the first paragraph of Section 6.2 repeated for each contour?

**Response:** The process described in the first paragraph of Section 6.2 is repeated for each contour. We have added one sentence in the first paragraph of Section 6.2.

6 How do you "filter out invalid parts and collapse multiple instances into single part" (last sentence of 6.2)?

**Response:** Notice that we could gathered all super-faces whose projections lie within the closed contour. For the contour section enclosing empty space, we could generate nothing. The invalid parts (generated with depth-discontinuous contour section Figure 8(a)) are identified by a group of nonadjacent super-faces. The invalid parts are discarded. The multiple instances of the same part (Figure 8(c)) are identified by the same sets of the super-faces. The repetitive instances are removed. We have added some scentences in the last paragraph of Section 6.2.

7 What is the sub-SFG (Section 7, "Multi-scale part suggestion")? What makes the suggested segments multi-scale? If the normalized cuts produces sets of faces, these faces may not have silhouettes, so how can the contour descriptor and distance be computed? More explanation is needed.

**Response:** 1) The sub-SFG is the SFG corresponding to the part suggested. We have reword the second scentence of the “Multi-scale part suggestion”in Section 7.

2) The retrieved part is future partitioned into several segments. We suggest segments according to the distance between the segments and the part corresponding to the query: given the scale factor $S$, the segments, whose distance to the part corresponding to the query is less than or equal to $S$ are presented as part suggestion. We have reworded the latter part of the “Multi-scale part suggestion” part of Section 7.

3) The contour descriptor is not used here. The distance is defined as the Euclidean distance between the OBB of the segment and the corresponding part. We have reworded the latter part of the “Multi-scale part suggestion” part of Section 7.

8 Are the whole-object contours represented with m=20 samples when computing distances (Section 8, "For the two kNN graph methods...")?

**Response:** No. Each of the whole-object contours is sampled under 3 different scales. Our levels place 50, 150, and 250 sampled points along the contour respectively. The contour descriptor is then computed for the sampled points. We have added 3 scentences for the second paragraph of the RC-kNNG retrieval performance part of Section 8.

9 I don't understand where the scale parameter S fits into the multi-scale part suggestion (Section 7, "Multi-scale part suggestion" and Figure 14). S does not appear in the expression for Tn. The figure shows different scale levels, but all shapes appear to be segmented into 2 or 3 parts uncorrelated with the scale level.

**Response:** The scale level is given by the user. The segments, whose distance to the corresponding part are less or equal to the scale level $S$, are suggested. We have reorganized the “Multi-scale part suggestion” part of Section 7.

10 Why is it a problem that the contour descriptor is not scale invariant (Section 9, Limitations) if the database stores contours at various scales?

**Response:** Although we have stores contours at various scales, there are some scales are missing.

11 Typos

Section 2: "develop a statistical models"; Section 3: "a online phase"; Section 6.3.2: "an other edge" should be "another edge"

**Response:** We have corrected the typos.

12 Why does increasing the superface count shrink the matched part (Figure 15)? There may be a normalization term missing somewhere.

**Response:** With the increasing of the superface count, the bigger superfaces are partitioned into smaller ones. Therefore, the extracted part shrinks.

13 The primary weakness is a missing evaluation of the fundamental premise. The premise of the approach is that the example database becomes larger by implicitly containing all possible segmented parts. This is interesting and a worthwhile idea, but it is unevaluated. Why not simply run segmentation on the database during preprocessing? Then any of the approaches mentioned in the related work (sketch-based or 3D shape-based) which retrieve similar entire shapes from a database could be compared to. Can users no longer find the sub-parts they are looking for?

**Response:** 1) I agree with you that the premise of nearly all the sketch-based shape retrieval methods is that the example database containing all possible shapes. We leave the evaluation of the fundamental premise for the future (See the future work part of Section 9).

2) We have made a comparison to the “pre-segmentation approach”; See the “Comparison” part of Section 8.

14 The "RC-kNNG Retrieval Performance" doesn't explain whether the alternative indexing structures retrieve the same parts.

**Response:** We have added the comparison among the alternative indexing structures; See the last paragraph of the “RC-kNNG Retrieval Performance”part of Section 8.

15 Does the proposed approach work on a different database than the one it was developed with?

**Response:** I develop the approach on a database including 73 shapes from different categories: humanoid, lamps, fourlegs, birds. The database is then extended to 513 shapes for evaluation and application.

**Review 4**

1 Clarity of Exposition: The exposition is fairly clear, but the number of camera views and how they are determined is not given.

**Response:** We extract contours from 21 camera views: 3 canonical side views, 4 corner views and 14 uniformly sampled views; See the “Camera views” part of Section 8.

2 The super-faces are analogous to super pixel representations in images — perhaps citation of a super pixel paper would be appropriate.

**Response:** We have cite a super pixel paper; See the third paragraph of the “Offline Phase” of Section 3.

1. — missing discussion of choice number and location of camera views needed. This seems to be a key issue, and would seem to depend on the types of objects in the data base.

**Response:** We have compared different camera views: 7 views, 21 views, 49 views, and 112 views; See the “Camera views” part of Section 8.

4 — the extreme simplicity of the partial shapes used. In a significant database an extraordinarily large number of partial geometries could match the very simple sketches used here.

**Response:** I agree that there could be an extraordinarily large number of partial geometries matching the very simple sketches. The ranking algorithm (introduced in Section 5.2) could rank the results according to the matching error. The contextual information could be adopted to assist our partial matching algorithm. We have noted this point in the “future work” part of Section 9.

5 — the very small size of the data based searched. There are obviously larger data sets available, making it seem suspicious that this doesn’t scale well.

**Response:** We have demonstrated our method on the database with 513 models. The performance is shown in table 1. I agree that it is better to test our algorithm on a larger database, we leave it for the future; See the “future work” part of Section 9.

6 In the example applications, the ideas are interesting, but the pose of the partial shape used for computing the boundary contours appear carefully selected.

**Response:** If the pose is not well selected, the visual cue is not so significant, our method could not provide the expected result. We have noted this point in the “future work” part of Section 9.

7 There are examples of results for a user sketch, but it isn’t clear whether this is just a sample “user sketch” made by one of the authors. Similar to FKS04, a compelling test would be to give the user something specific to design from pieces of objects in the data base.

**Response:** Good suggestion! We have added the application “Photo-driven part-based modeling” in Section 7.