Learning Visual Composition Preferences from an Annotated Corpus Generated through Gameplay

Reid Swanson Dustin Escoffery Arnav Jhala
Baskin School of Engineering
University of California Santa Cruz
Santa Cruz, California 95064
Email: {reid, descoffe, jhala}@soe.ucsc.edu

Abstract—This paper describes a game called Panorama, designed to facilitate data collection to study visual composition preferences. Design considerations for Panorama, implementation of composition rules, and data collection for an experiment to learn individual and collective preferences is described. Images taken through gameplay in Panorama are automatically scored for composition quality and contribute to a corpus of domain-specific virtual photographs annotated by visual features and scores. Scores in Panorama represent rules of good composition from photography textbooks. In the current version, Panorama scores photographs along balance, thirds alignment, symmetry, and spacing dimensions. Pairwise preference rankings are collected on images from this corpus through crowd-sourcing. Results are presented from data on relative pairwise rankings on the images to learn individual as well as general composition preferences over features annotated in Panorama images. This work seeks to extend the ability of AI systems to learn and reason about high-level aesthetic features of photographs that could be utilized for various procedural camera control and aesthetic layout algorithms in video games.

I. INTRODUCTION

Much of the information around us is in visual form, be it in charts, layouts of web-pages, photographs from vacation trips, video lessons from online classes, or video games for entertainment and education with rich 3D environments. High-quality cameras are now available on cheap portable devices and the average person is not only a consumer but a significant producer of visual information. It becomes increasingly important to understand the aesthetics of visual composition as we generate more content that is not crafted by experts. Beyond the understanding of our visual aesthetic preferences, this will enable computational tools that could enhance the creativity of average users to produce high quality visual content.

Photographic composition evaluation is also becoming an important aspect of 3D graphical games and simulations. Complex graphical environments require intelligent virtual camera placement algorithms, particularly ones that emphasize visual storytelling through the immaculate placement of objects in a scene to manipulate the viewer's perceptions. As the richness and complexity in virtual environments, such as interactive storytelling and video games increases, so too does the need for complex adaptive cinematic camera control. While current constraint-based approaches [1] are efficient in searching through a space of camera viewpoints to find the best view that matches viewing constraints, the solutions are not guaranteed to follow photographic composition rules.

Increasing quality and processing power of digital cameras is leading to large collections of interesting user generated photographs and a broader interest in better photography and videography. Recent advances in the field of Computational Photography [2], [3] have led to several automatic configuration features in digital cameras to help amateur photographers take high quality pictures without worrying about camera parameters like focus and exposure. So far, no such tools exist that can assist users with better composition of their photographs, which relates to the task of determining cropping boundaries to include/exclude and frame subjects within photos. This is a challenging problem because composition depends on various factors including the subject matter of the picture, communicative intent of photographers, and their composition preferences. None of these factors can be easily explored through current vision-based algorithms of photo evaluation.

For understanding visual composition [4], we need to find models to compute the overall composition quality of an image as well as capturing the preferences of individual viewers. There are two ways in which this problem has been approached. Researchers [5] have created mathematical approximations of common rules of composition like visual balance and rule-of-thirds alignment [5]. These rules are then used to evaluate frames and camera directives are used to move, pan, rotate, and tilt the camera to achieve optimal score. This approach is useful for virtual in-game cameras where the information in the frame is available to the system but is not practical for real cameras. Another approach is to apply vision filters to segment the image into parts and then evaluate each part on pre-scripted composition equations [3]. This approach is computationally expensive due to the variety and complexity of image processing filters and relies on vision algorithms for object detection and segmentation during the preprocessing stage. This approach also uses fixed composition equations but can incorporate more optical features that are available in the raw image data.

One goal of this project is to identify visual features that contribute to the compositional quality score of an image. Another is to assess their relative importance in predicting aesthetic quality perceived by individual users and generally across users. Given an image, there is often disagreement...
among individuals on goodness of composition due to their individual preference for visual features as well as their interpretation of the subject matter. To obtain our initial feature set we started with visual guidelines that were discussed in beginner-level photography textbooks. We represented several visual features including (balance, rule-of-thirds, symmetry and spacing) that were previously represented in work on computational evaluation of photographs and were also described in photography books. These features were used to score images taken in a game.

II. CHOICE OF VISUAL REPRESENTATION

If we are to use preference rankings from people to assess the quality of a photographic composition. We will not know whether an individual liked (or disliked) an image because of the placement of symbols in the frame, or if it is due to some particular quality of the image that personally relates to that individual. To minimize these issues we decided to use gray-scale line drawings to render our virtual photographs. At one extreme we could have used abstract symbols, such as circles, squares and triangles. However, we still wanted to maintain some tie to real world photography, which we believe would allow us to obtain higher quality user rankings and hopefully better translate to actual photographs.

There are two problems we address with the choice of representation in Panorama. First, we need a tool for collecting many images with similar visual properties but variation in composition and easy to extract features. This avoids use of image processing techniques to pre-process images and images with varying intensities, lighting conditions and other factors that are present in images available from the web. In addition to this, real images have meaningful objects that affect individual perceptions on goodness of the image. It is also impractical to have many subjects take pictures of the same object or location in the real world for data collection. Panorama’s simple theme focuses users on the layout and placement of objects in the frame during comparisons. The second problem is clear demonstration of this framework for learning both the individual and group composition preferences over several image features. We address this from a corpus of annotated images generated from Panorama and obtaining pairwise comparisons through the crowd-sourcing platform Amazon’s Mechanical Turk from a general audience. In our pilot experiment, analysis reveals significant effect of various visual features like balance, and rule of thirds, that are described in beginner-level photography books, on user preferences for composition. As a first step toward a data-driven analysis of compositional features, our results are encouraging. As other types of data, such as numerical likert-scale rankings of individual images, becomes available we should be able to leverage different preference learning models, such as collaborative filtering, to derive an even better understanding of people’s compositional preferences.

III. RELATED WORK

There are two areas in which work related to this project has been carried out. In the area of intelligent virtual camera control many algorithms incorporate frame evaluation for ranking shots while searching for the best camera position given a set of visual constraints [5], [1]. Bares proposed the use of photographic composition rules of Balance, Emphasis, and Layout to implement an intelligent assistant that suggested camera movements for better composition based on these rules. Chang and Chen [6] employ photographic composition in a search problem, and describe a method for finding good compositions in panoramic images. In their work, quality is not a function of pre-defined heuristics, but is measured statistically from a database of expert photographs. Images are analyzed for structure and visual saliency in a computer vision step in order to find similar candidates. They use stochastic search to find images most similar to expert examples. Liu et al. [8] discuss a technique for cropping images to computationally improve composition. They combine several of the aforementioned techniques for a practical demonstration. They do a pre-processing vision step similar to Chang and Chen [6] to detect regions of visual saliency. The regions are scored for rules similar to Abdullah et al. [7], and likewise use particle swarm optimization to suggest crop parameters.

Banerjee et al. [9] describe an application of photographic composition in digital photography. They describe an algorithm that applies composition rules to automatically align the subject and shift the focus in a digital camera. Their system can easily be improved as new composition heuristics are discovered. Lok et al. [2] discuss how composition can inform the layout of user interfaces. However, unlike photographic composition, they are more concerned with concepts like visual balance, and less about photography principles like rule of thirds. Su et al. [10] propose a framework for providing composition recommendations, using image analysis and machine learning. They construct a personal preference model in an offline learning step, in which users specify preferred photographs of scenic landscapes. Instead of composition rules, they employ a bottom-up approach for extracting thousands of features, and selecting relevant ones by boosting. A limitation of their system is finding the appropriate image dataset. Panorama circumvents this problem by generating its own dataset. Ke et al. [11] suggest a set of high-level image features to use for classification. Their features were inspired by the opinion of professional and amateur photographers in addition to photography literature. They provide mathematical formulations to calculate the features from real photographs. Using Naive Bayes, they show that high-level features can successfully classify professional photographs. Moorthy et al. [12] describe an experiment for estimating aesthetic appeal of videos. They collect user rankings for video appeal, and train support vector machines to classify videos as good or bad. They hierarchically construct video features from individual frame calculations. Yannakakis [13] suggests a protocol for preference learning that is sensitive to personal affect.
In his system, pairs of alternative choices are presented and evaluated by questionnaire. His model uses a four-alternative forced choice (4-AFC) rating system for generalizing across different subjects. This project adopts the 4-AFC protocol of pairwise comparisons, by collecting image preference data in 4AFC format, to construct the most general model of quality.

**IV. GAME DESIGN AND IMPLEMENTATION**

Panorama (see Figures 1 and 2) is a 2D sidescrolling game developed on the Microsoft XNA framework. The primary mechanic of the game is of taking pictures with a camera, similar to games like Afrika [14]. Through gameplay, players take pictures of a scrolling panorama. These pictures get added to the individual player’s library along with composition scores calculated by the system.

We decided to use line drawings and initially only have grayscale images. Although abstract art could have removed bias related to specific objects represented in images, the transfer of preferences collected through this experiment to real images would have been challenging. For this reason we decided to include simple but recognizable objects in our level design. These images have well-defined visual features and annotations for various derived aesthetic features from the graphical representation of the objects in the game. Images from this corpus can now be used to collect preference rankings that enable us to determine importance of recorded features to an individual’s model of composition quality. The algorithm in turn, can improve the scoring in the game to adapt to individual players.

The main objective of the player is to maximize their composition score on pictures. There are four scores that are given to a player playing the game. After each picture is taken, the player gets a badge {Gold, Silver, Bronze, Red} for the quality of their picture along each of the three composition parameters, Balance, Thirds, and Spacing. They also get an overall score as a text feedback labels {Awesome, Great, Good} based on their cumulative score for the three parameters. There is no other feedback given to the player on how to improve their score (i.e. how to improve balance).

The design of Panorama addresses two challenges that were highlighted in the earlier section with current approaches to photographic composition evaluation and modeling. First, Panorama levels are made from simple primitives. Some of the subjective bias and context from real-world pictures is removed because all players take pictures in the game either to maximize their score or to take pictures that will be popular among other players. Second, the scoring system allows players to pick the dimension along which they prefer optimizing rather than worry about conflicting goals (such as rule of thirds conflicting with emphasis on a particular object). Finally, each level adds a visual feature so data collected from panorama will provide insight on how each visual feature contributes to composition preferences in individual players and in general over the entire subject pool.

**V. COMPOSITION EVALUATION: VISUAL FEATURES IN PANORAMA**

During gameplay, when a user takes photographs through the in-game virtual camera, seven features from the game are recorded in the database as annotations for each image. These are listed in Table I. Of these, four features are computed in Panorama as scores. The balance and thirds features are based on earlier work by Bares et al [5]. Spacing is a feature that is introduced based on photography rules. Overall score is the cumulative score of these three features.

**Center of Mass (M)**: Given that the origin is at the center of the frame that contains i objects, Let $a_i$ and $c_i$ be the area...
TABLE I: Summary of features recorded for each image in the game. Size represents number of variables that represent sub-parts of evaluation.

<table>
<thead>
<tr>
<th>Feature</th>
<th>Size</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Symmetry</td>
<td>S</td>
<td>Distance from scene’s center of mass to four axes: ( x = 0, y = 0, x = y, ) and ( x = -y ).</td>
</tr>
<tr>
<td>Thirds</td>
<td>T</td>
<td>Average distance of each objects centroid to the nearest thirds line on both axes.</td>
</tr>
<tr>
<td>Spacing</td>
<td>Q</td>
<td>Binary indication that a region of space is not empty, as partitioned by a quadtree of depth four.</td>
</tr>
<tr>
<td>Objects</td>
<td>N</td>
<td>Number of objects.</td>
</tr>
<tr>
<td>Crops</td>
<td>C</td>
<td>Number of objects cropped by the edge of the frame.</td>
</tr>
<tr>
<td>Occlusions</td>
<td>O</td>
<td>Number of objects that intersect another object.</td>
</tr>
<tr>
<td>Size Ratio</td>
<td>R</td>
<td>Area of the smallest object, divided by area of the largest object.</td>
</tr>
</tbody>
</table>

and centroid of the \( i \)th object respectively.

\[
M = \sum \frac{a_i c_i}{\sum a_i}
\]

The Symmetry feature \( S \) is calculated using the center of mass \( M \), and the width and height of the frame \( w, h \). Overall symmetry of the frame is calculated along 4 dimensions. Along the \( x \) axis, along \( y \) axis, and along the two diagonals. Equations for each of these are shown below.

\[
S = (s_{x=0}, s_{y=0}, s_{x=y}, s_{x=-y})
\]

\[
s_{x=0} = 2 \frac{|M_x|}{w}
\]

\[
s_{y=0} = 2 \frac{|M_y|}{h}
\]

\[
s_{x=y} = 2 \sqrt{\frac{2(|M_x-M_y|)}{w^2+h^2}}
\]

\[
s_{x=-y} = 2 \sqrt{\frac{2(|M_x+M_y|)}{w^2+h^2}}
\]

Thirds alignment

Let the function \( \delta \) compute the normalized distance of point \( p \) to the nearest thirds line along axis \( a \), where \( F_a \) is \( w, h \) for \( a=x, y \). The degree of thirds alignment \( T \) is calculated using the average distance across all elements centroids.

\[
\delta(p, a) = 3 \min_{t: a = \pm 1/6} |\frac{p_a}{F_a} - t|
\]

\[
T = 1 - \frac{1}{2n} \sum (\delta(c_i, x) + \delta(c_i, y))
\]

Spacing

The spacing feature is calculated using a quadtree decomposition of the space contained within the frame. Quadtree is a data structure that contains the whole frame of the picture as the root and adds nodes by recursively subdividing the frame into four parts for three levels. The spacing computation takes into account filled nodes in the quad tree data structure (see Figure 3). This is an efficient way of calculating spacing as a completely empty frame only has to be evaluated once.

Let the function \( q_0 \) indicate whether some element \( e \), in the set of all elements \( E \) included in the frame, intersects a given rectangle \( r \). The function \( q_{n+1} \) recursively divides rectangle \( r \) into four quadrants: top-left \( r_{tl} \), top-right \( r_{tr} \), bottom-left \( r_{bl} \), and bottom-right \( r_{br} \). The Spacing feature \( Q \) is the string of bits generated from the quadtree function \( q_3 \) for frame rectangle \( F \).

\[
q_0(r) = \begin{cases} 1 & \text{if } \exists e \in E : e \cap r \neq \emptyset. \\ 0 & \text{otherwise}. \end{cases}
\]

\[
q_{n+1}(r) = (q_0(r), q_n(r_{tl}), q_n(r_{tr}), q_n(r_{bl}), q_n(r_{br}))
\]

\[
Q = q_3(F)
\]

The Objects feature \( N \) is the number of elements in set \( E \), normalized by \( 1/10 \).

\[
N = \min(1, \frac{|E|}{10})
\]

The Crops feature \( C \) is the number of elements in set \( E \) that are not fully within frame rectangle \( F \).

\[
C = \min(1, \frac{|\{e : e \cap F \neq e\}|}{10})
\]

The Occlusions feature \( O \) is the number of unique intersections between two elements in \( E \).

\[
O = \min(1, \frac{|\{e_1 \cap e_2 : e_1 \neq e_2 \wedge e_1 \cap e_2 \neq \emptyset\}|}{10})
\]
The SizeRatio feature $R$ uses the minimum and maximum of all element areas $a_i$.

$$R = \frac{\min_i a_i}{\max_i a_i}$$

VI. USER STUDY

To assess the ability of our features to model the stylistic preferences we performed a user study of images collected from the Panorama game. We first collected 20 images each from five individuals who played through the same level of Panorama for a total of 100 virtual photographs.

We obtained a large number of preference rankings from several dozens of participants recruited through Amazon’s Mechanical Turk crowd-sourcing platform to obtain pairwise preference rankings of these images. There are $\binom{100}{2} = 4,950$ possible ways of combining these 100 images as pairs. We randomly sampled approximately half of these pairs (2,470) to use in our preference rating experiment. For each pair one image was placed on the left hand side of the survey and clearly marked image (a). Image (b) was shown directly to the right by about 20 pixels. Below the pair of images, we asked the participants to select the image in which they thought the objects were better composed and arranged in the frame. They were given four possible choices to rate the pair of images.

1) Image (a) is better
2) Image (b) is better
3) Image (a) and (b) are equally good
4) Image (a) and (b) are equally bad

Since the graphics in these images are simple line drawings that are not as sophisticated as the types of graphics one might be more familiar with in popular media sources, we specifically asked the users to try to avoid rating the images based on the quality of the graphics themselves. Using this methodology we were able to collect rankings from a total of 36 participants of which 31 provided at least 12 rankings.

From these rankings, we created a training and test set from the 24,689 pairwise preference rankings we collected from our participants by randomly assigning each example to one of these categories, such that the training data had approximately 80% of the total examples. This resulted in a training set of 19,749 examples and a test set of 4,940.

VII. RESULTS AND DISCUSSION

In our first experiment we examined whether we could use this data to learn a model to accurately predict the preference rating of an arbitrary user on an unseen pair of images. Note that either of the images or both could have been seen in the training data, but not together as a pair. We used an available multi-class Support Vector Machine learning algorithm [15] with a Radial Basis Function(rbf) kernel, which has been shown to work well across many different types of problems. To learn the hyperparameters ($C, \gamma$) we performed a grid search that optimized for the weighted average F-Score of the class labels using a 5-fold cross-validation on the training data. The weighted average F-Score is computed by summing the F-Score for each class multiplied by the relative frequency of that class in the test data. In our data there were 2,180 test examples labeled as $A > B$, 1,412 labeled as $B > A$, 766 labeled as $A = B$ (Good) and 582 labeled as $A = B$ (Bad).

A. Global Preferences

Figure 4 shows the learning curves when training a model with the specified number of examples and evaluating that model on the entire test data set. The classifier is able to reach its peak performance relatively quickly compared to the total number of training examples we have collected, although it still needs several thousand comparisons to do so. There is also a large discrepancy in performance between the different types of preferences. Some of this can be explained by the relative amount of training data available for each class. However, the curves also seem to indicate that the cases in which there is no clear preference are more difficult to learn and barely reach chance levels of performance. Overall, the weighted average performs much better than chance, but is still relatively low.

The curves in Figure 4 indicate that the classifier has considerable difficulty in learning when a user does not have a clear preference for an image one way or another. To get an upper bound on the performance of our classifier, we created a new training and test data set that only kept instances in which a user had a clear preference for one of the images over the other. The learning curves of this classifier are presented in Figure 5. We treat this as a standard binary classification task, although there is no reason to treat $A > B$ as a positive rather than $B > A$ as a positive example. For this reason we also plot the performance of each class separately and include the weighted average.
B. Individual Preferences

Although we are able to make relatively accurate predictions of the preferences of an arbitrary user, we would also like to be able to model and predict the preferences of specific individuals. This is important because preference rankings are highly subjective that often do not have universally correct answers. What may be aesthetically appealing to one user may be the exact reason another user finds the image unappealing. To assess how well we can model individual users we performed another set of experiments.

Analogous to the multi-class experiment described above we created a separate learning curve for each individual using only their training and testing data. Figure 6 shows the mean values of the learning curves over the 31 individuals for each class and the weighted average, which includes 95% confidence intervals. The widening of the confidence intervals as the number of examples increases is a result of fewer participants that provided a large number of comparisons. The graph shows that we are actually able to make individual predictions at greater accuracy and with many fewer training examples. This suggests that individuals prefer different images from each other, which leads to conflicting rankings. This is a source of noise in the training data for arbitrary users resulting in lower overall performance. The performance of ambivalent (A=B) preferences is still comparatively low, but the curves indicate there is room for improvement given more examples.

Figure 7 shows the analogous learning curves when we remove rankings that did not show a clear preference for one of the images over the other. Again, we are able to achieve much higher performance for an individual user than we can for an arbitrary user by aggregating all of the data together. We also need far fewer training examples to achieve significantly

Fig. 5: Learning curve over the entire dataset for binary classification for clear preference for one image.

Fig. 6: Mean values of the learning curves over the 31 individuals for each class and the weighted average, which includes 95% confidence intervals.

Fig. 7: Learning curve for individual participants of binary classification for clear preference for one image.
greater than chance results and only about 500 examples to reach near peak performance.

Our analysis indicates that we are able to predict individual preferences for a pair of images based on our features. While we believe these results show that standard compositional rules can be used to rank known images with high accuracy, we would also like to show that these features are useful in ranking unknown images as well. To assess how well our features predict a preference for two unknown images we performed an evaluation similar to a leave-one-out cross-validation. For each test example, we identified the two images that were being compared and removed all training examples containing either of those two images. We trained a model from this reduced set of training examples (165 on average) and made a prediction for the single test example. We then computed the accuracy for a user by taking the number of correct predictions divided by the number of test examples.

The average accuracy of all participants using this evaluation was 0.608 ± 0.048. Although the accuracy is a substantial decrease from our best results, the performance is still significantly better than chance and demonstrates we can make relatively accurate predictions on completely unseen images using our features.

C. Analysis of feature set

As a final analysis we performed a deeper examination of our feature set. To do this we used the feature selection method of [16] to find the top 10 most important features for each user. Figure 8 shows the frequency of the 23 features that were in the top 10 across all participants. For example, feature 54 (ΔROT - Rule of Thirds), was among the top 10 most predictive features for 14 participants. We also examined the feature profiles of users based on the number of rankings they provided us. We believed the number of rankings could be an underlying indicator of the level of interest in the task, which might result in distinct feature profiles for the different groups. In Figure 9 we present the feature profiles for users in the top quartile based on the number of comparisons. The feature distributions between all pairs of quartiles are significantly different (p < 0.05), except 3 and 4, using Fisher’s exact test. One interesting observation is that feature #184 which represents the ratio of size of objects in the image appeared among top features in the first quartile of users but not in other ones. On the other hand thirds alignment feature #54 was among the top features in the bottom three quartiles and not in the first.

VIII. Conclusion and Future Work

We have described a framework called Panorama, a game for collecting images that are automatically annotated by various visual features related to composition. Images from Panorama are collected in a database and rated with pairwise composition preferences by a large number of users. This provides us with data to learn composition preferences for individuals as well as general principles.
We have shown that the features investigated in the Panorama game can predict composition preferences of specific users as well as general quality of photographic composition. Not only can we use these features to characterize and predict known images, but we have shown that these features are sufficient to rank unseen images with high accuracy. This is the first step in establishing a workflow of data collection and analysis to systematically collect and analyze data for various features of the visual system. Future work would involve analysis of more data to build robust models of composition preferences. Panorama is designed such that each level in the game adds new visual features. The images in the database analyzed in this paper were taken from the first two levels and included straight lines and grayscale values. Later levels include variations in hue and saturation of colors and levels that span the color gamut. Once data collected from the early levels is analyzed, color, exposure, shadows, etc. will be added to Panorama levels and results at each level compared to get better models of general photographic aesthetic preferences. We are currently investigating application of this model to predict composition quality of photographs of real scenes. We have already integrated the features into a mixed-initiative photo cropping tool.

We have taken the equations from the Panorama game and implemented them in an application called SmartCrop. SmartCrop takes images taken from real cameras along with annotations for important objects in the scene (bounding boxes)(see Figure 10). It then takes 1000 samples of various frame sizes across the picture and selects the top 10 crops that include a given object in the scene. Detailed results of the SmartCrop user study are outside the scope of this paper, but results on natural panoramic pictures with SmartCrop against human image crops showed promising results. Detailed analysis of this work is being carried out. SmartCrop application, that incorporates the composition equations from Panorama and applies them to real photographs could be integrated with vision-based algorithms where the results of Panorama can be applied to real images. Other immediate applications of this work include virtual camera control in games, machinima generation tools, pre-visualization storyboarding assistants, and even in aesthetic evaluation of abstract visuals such as graph layouts.

**TABLE II:** The feature names corresponding to the feature id in the previous figures. For the quadtree indices refer to Figure 3.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Feature Id</th>
</tr>
</thead>
<tbody>
<tr>
<td>∆Q411</td>
<td>3</td>
</tr>
<tr>
<td>∆Q142</td>
<td>15</td>
</tr>
<tr>
<td>∆Q44</td>
<td>45</td>
</tr>
<tr>
<td>∆Q311</td>
<td>48</td>
</tr>
<tr>
<td>∆ROT</td>
<td>54</td>
</tr>
<tr>
<td>∆Q431</td>
<td>66</td>
</tr>
<tr>
<td>∆Q31</td>
<td>111</td>
</tr>
<tr>
<td>∆Q423</td>
<td>120</td>
</tr>
<tr>
<td>∆Q24</td>
<td>126</td>
</tr>
<tr>
<td>∆Q133</td>
<td>141</td>
</tr>
<tr>
<td>∆Q334</td>
<td>144</td>
</tr>
<tr>
<td>∆Occlusions</td>
<td>159</td>
</tr>
<tr>
<td>∆Q23</td>
<td>168</td>
</tr>
<tr>
<td>∆Size ratio</td>
<td>184</td>
</tr>
<tr>
<td>A:Q1</td>
<td>256</td>
</tr>
<tr>
<td>∆Q1</td>
<td>258</td>
</tr>
<tr>
<td>A:Q0</td>
<td>259</td>
</tr>
<tr>
<td>∆Q0</td>
<td>261</td>
</tr>
<tr>
<td>A #: Objects</td>
<td>274</td>
</tr>
<tr>
<td>∆ #: Objects</td>
<td>276</td>
</tr>
</tbody>
</table>

**References**


