A Geodesic Active Contour Framework for Finding Glass

Kenton McHenry[1] and Jean Ponce[1,2]
[1] Beckman Institute
University of Illinois at Urbana-Champaign, USA
[2] Département d'Informatique
Ecole Normale Supérieure, Paris, France

Abstract. *This paper addresses the problem of finding objects made of glass (or other transparent materials) in images. Since the appearance of glass objects depends for the most part on what lies behind them, we propose to use binary criteria ("are these two regions made of the same material?") rather than unary ones ("is this glass?") to guide the segmentation process. Concretely, we combine two complementary measures of affinity between regions made of the same material and discrepancy between regions made of different ones into a single objective function, and use the geodesic active contour framework to minimize this function over pixel labels. The proposed approach has been implemented, and qualitative and quantitative experimental results are presented.*

1. Introduction

Material classification is important in practical computer vision applications as well as general object recognition [1]. Materials have several properties that can be picked up visually such as surface reflectance [17] and texture [24]. Texture is often sufficient to distinguish different opaque materials [5, 11]. On the other hand, transparent objects are characterized by the fact that one can see through them, rather than by any intrinsic texture pattern. The main focus of the machine vision research dealing with transparent objects has been concerned with 3D structure from X [4, 9, 15, 16, 22] and layer separation in the case of reflections [12, 23]. Relatively little work has been concerned with the automatic identification of objects made of transparent materials [18]. Adelson et al. [2] have shown that X-junctions —where a transparent material intersects a color/texture boundary— may play an important role in this process, and given rules for classifying junctions as being due to a transparent boundary or not. This has been applied to the separation of transparent overlays in [21]. McHenry et al. [14] have shown that edges across glass objects' boundaries[1] show up quite reliably in the output of the Canny edge detector [8], and proposed various cues to capture the transparent, refractive and reflective properties of glass across its boundaries, and classify the corresponding edges. In this approach, the active contour method of Kass et al. [10] is then used to identify a single glass region enclosing the identified edges.

[1]We will use "glass" in the rest of this presentation to designate transparent, highly reflective and refractive materials such as glass.

This paper describes an alternative, region-based approach to the problem of segmenting images into glass and non-glass components. By focusing on regions, we alleviate the need for a final grouping stage as needed in an edge-based method, and thus do not have to assume that only one glass object is present. Our method can be outlined as follows: Since the appearance of glass objects depends for the most part on what lies behind them, we propose to use binary criteria ("are these two regions made of the same material?") rather than unary ones ("is this glass?") to guide the segmentation process. After an initial set of homogeneous image regions has been identified by a traditional segmentation algorithm [7], pairs of regions are related by an *affinity* measure, providing some indication as to how well two regions belong to the same material; and a *discrepancy* measure based on how much one region looks like a glass-covered version of the other, providing a basis for separating regions made of different materials. We combine these two complementary measures into a single objective function. Though optimizing this function is a combinatorial problem, we show that it can be modified to fit into the geodesic active contour framework [6], allowing a solution to be found efficiently.

The rest of the paper is organized as follows. Section 2 describes the initial image segmentation process and the proposed measures of discrepancy and affinity. In section 3, we present a cost function derived from these measures and effective means for optimizing it in a geodesic active contour framework. We present our experiments in section 4, and conclude with a brief discussion in section 5.

2. Characterizing Glass Regions

In [14], rectangular samples are taken from opposite sides of small edge snippets from the output of the Canny edge detector and compared using a series of cues associated with characteristics of glass: transparency, refraction and reflection. The cues operate on the assumption that one sample should be some arbitrary piece of background and check whether the other looks like a glass-covered version of that piece. The two regions should look similar in terms of color, and somewhat similar in terms of texture. Since glass is refractive, we might expect the texture to be slightly different due to distortion. Because of reflection, we might also ex-

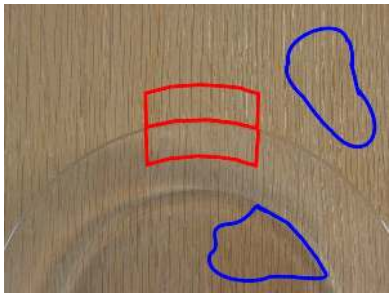

Figure 1. *In [14], region samples around edge snippets are compared in order to classify the snippet as glass or not glass based on similarity and distortion between them. Here, our goal is to check whether one region is a glass-covered version of another, and we may potentially compare any two (blue) regions in the image.*

pect to see specularities as well.

While these cues hold most strongly at edges between adjacent regions, the same cues can in fact be applied between any pair of image samples (Figure 1). Rather than focusing on rectangular region samples along image edges, we use in this paper the homogeneous texture/color regions found by the graph-based segmentation algorithm of Felzenszwalb et al. [7] as an initial partition of the image. These regions that are then compared as explained in the rest of this section to identify glass/non-glass pairs. It is important in our case that the initial regions provide an *over* segmentation of the image, which is easily achieved by appropriately setting the parameter that governs the behavior of the algorithm described in [7]. In addition, we have modified the measure of texture/color discrepancy between pixels proposed in [7] to add weight to edge pixels and prevent regions from bleeding over Canny edges, which have been shown in [14] to be reliable cues for glass boundaries. Finally, small and thin regions are merged with their neighbors.

Ideally, given these initial regions, we would like to define a single homogeneity measure expressing how much regions from the same material belong together. Since we are separating glass from non-glass, this means giving high values to regions within glass, high values to regions not within glass, and low values to regions when one is in glass and one is not. As explained below, this does not seem possible, due to the optical properties of glass and other transparent materials. We propose instead to quantify how much pairs of regions don't belong together, or do, using two separate *discrepancy* and *affinity* terms.

2.1. Discrepancy

McHenry et al. propose in [14] to find glass edges using a classifier trained on the following cues between two regions incident to the same edge, assuming one is a glass-covered version of the other:

- **Color Similarity (C):** the color tends to be similar in both regions.
- **Blurring (B):** the texture in the glass region is blurred.
- **Overlay Consistency:** the intensity distribution in a glass region is constrained by the intensity distribution of its background by alpha **(A)** and emission **(E)** values.
- **Texture Distortion (D):** the texture of the glass region is slightly different due to refraction.
- **Highlights (H):** glass regions often have specularities.

We borrow from this work, and refer the reader to [14] for details. We do not use the blur cue here since it relies on the discrete cosine transform which is appropriate for rectangular samples but not for arbitrary region shapes.

The five cues (A, C, D, E, H) together provide us with a five-dimensional space to characterize the *discrepancy* between two regions —that is, a measure of how these regions are unlikely to come from the same material, in terms of one of them looking like a glass-covered version of the other. Rather than train a classifier in this five-dimensional space, the cues are grouped into three different classifiers of three terms each: ECA, DCA, HCA, and their outputs combined using a fourth linear support vector classifier which effectively weights the reliability of these sub-classifiers.

It has been shown in [19] that the value returned by a support vector classifier —that is, the signed distance of a data point to the dividing hyperplane— can be used as a probability with a proper mapping. This is done by fitting a sigmoid to the training data, which tends to give the positive training data very high probabilities and the negative ones very low probabilities. The range of values for which a test point can take on an intermediate value is small, and values increase quickly. This is fine when the classifier is rarely wrong, but for our purposes it is not desirable to have most of the assigned probabilities at the far extremes. In addition, our setting does not require discrepancy (or for that matter, affinity) measures that can be interpreted as probabilities, so we drop the log function and simply scale the SVM output so that on the training data the minimum output is 0 and the maximum is 1. This will be our discrepancy measure D_{ij} between regions i and j.

D_{ij} can be thought of as a distance separating regions depending on how much one looks like a glass-covered version of the other. If this is the case, a high value is returned. When the two regions look completely different, a low value is returned. Unfortunately, a low value in no way means that the two regions are both glass or both background. Consider the four regions shown in Figure 2. Let us call the two background types A, B and their glass-covered versions A', B' respectively. In this example, we expect $D_{AA'}$, $D_{BB'}$ to be high and D_{AB}, $D_{A'B'}$ to be low since they look very different. But what of $D_{AB'}$ and $D_{A'B}$? These pairs also look very different so will have low values, possibly lower values than D_{AB}, $D_{A'B'}$ depending on the background appearances. Thus, if we try to separate the regions into two groups, glass and non-glass, based on this

Figure 2. *Comparing two background regions to their glass-covered versions we expect to get a high discrepancy value. We will get low values when comparing the two background regions, the two glass regions, and the glass regions to the background region that does not correspond to it. This will allow for two possible segmentations, the correct glass/background segmentation and another with one glass region and the background that is not its own.*

measure alone, we may have many solutions. In the above example, both the correct segmentation (AB)(A'B') and an alternative (AB')(A'B) are likely just as good. In fact, we see that we can really only trust D_{ij} when its value is high.

2.2. Affinity

Since glass edges tend to show up reliably in images [14], it is reasonable to try and use them to relate regions. In particular, when two regions are connected by an edge, and they are on the same side of that edge, they can be correlated by some *affinity* value A_{ij}. Determining whether a path exists between two regions is a simple matter of walking along the edgels that come near one region's border and following the edge to see which other region borders it comes close to. When there is more than one way to approach another region, we consider only the shortest path. The process of finding all the connecting paths can be implemented efficiently by converting the edge output into a graph where the nodes are edgels that are junctions (more than two neighbors) or ends (one neighbor), and the graph edges are the edgel paths connecting these nodes (possibly empty). In our implementation, we consider a region to be near an edgel when it has a point less than four pixels away. This too can be done efficiently by simply dilating each region by four pixels.

We would like the value of A_{ij} to be high when two regions come from the same material (i.e., glass or non-glass), and low otherwise. To do this, let us consider what can happen on the edge path between two regions. If this path is rough, with many large changes in orientation, there is a good chance that it is going through a textured region, possibly merged by smoothing and hysteresis, or has left one object and entered another. On the other hand, if the edge path is smooth, there is a good chance we are following a single object's contour. Thus we define the affinity as

$$A_{ij} = 1 - \frac{a_{ij}}{\pi},$$

where a_{ij} is the maximum angle between consecutive edgel tangents on the path from region i to j. Straight paths will receive a value of 1, and paths with sharp turns will receive lower values. If there is no path between two regions, A_{ij} is set to 0. Edge detectors are known to behave oddly at junctions, possibly breaking one object's contour and connecting it to that of another. In many of these cases, an abrupt turn will result, thus it is beneficial to down weight sharp turns indicating our lack of confidence that they are true connections (Figure 3).

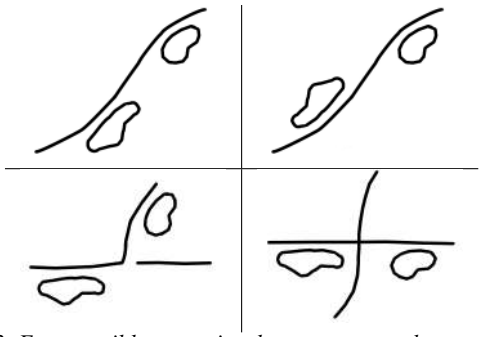

Figure 3. *Four possible scenarios that may occur when connecting two regions by an edge path.* **Upper left:** *Both regions are on the same side of a smooth edge path.* **Upper right:** *The regions are on opposite sides of an edge path.* **Lower left:** *Both regions are on the same side of a non-smooth path.* **Lower right:** *Both regions are on the same side of smooth path that is intersected by another path.*

One may ask what happens when a background contour intersects a glass object. Would such an occurrence strongly correlate glass and non-glass regions? Due to the highly refractive nature of glass, most contours passing behind a glass object will appear broken (Figure 4). In turn, this will force a sharp change in direction on any joining paths between the two regions, resulting in a low affinity. Of course, under the right conditions —perhaps a flat object like a glass plate lying on top of a striped table— such intersecting contours will result in an incorrectly high affinity. We can spot such hazards with the edge graph already built. When the path between two regions comes across an intersection which two high correlating paths could pass through, for example two paths at right angles,[2] we can do one of two things. Following [14], we could evaluate each of the good paths through the junction and only allow information to pass along the most glass-like path, or just prevent all information from passing through the junction. In our experiments we do the latter. We also extend the notion of affinity to pairs of regions that don't have a direct edge path between them, but are connected through several edge fragments along adjacent regions.

Figure 5 shows typical examples of affinity strengths between sample regions and all others. Let us point out that there is a second advantage to glass having a high refractive component in that the area within and just outside a

[2]The topology of an edge detector's output is unreliable in most circumstances. However in the case of glass, it is a viable means of relating regions.

Figure 4. *Checking to see whether two regions are on the same side of a smooth edge is a good way of linking together regions that are both glass or both not glass. Edges that could incorrectly link glass and background regions by passing behind the glass object are usually broken by refraction, thus making their edge path non-smooth.*

glass object's contour tend to have the glass boundary as the longest smoothest path around. Thus glass regions tend to be better correlated to one another than background regions are to each other.

We can think of A_{ij} as a measure of affinity between two regions, giving high values to regions within the same contour. However, a low value does not necessarily mean two regions belong in different groups. In fact in most cases it will mean that there is no edge path between them. As before, we can only really trust A_{ij} when its value is high.

3. Combining the Measures

We have been calling D_{ij} and A_{ij} discrepancy and affinity measures, but they may be better thought of as measures of certainty of discrepancy and affinity: When the discrepancy is 1, the regions compared are as likely to consist of different materials as can be inferred from the training data. On the other hand, when it is low, we cannot ascertain whether one region is glass, and the other is a piece of background. Similarly, when the affinity A_{ij} is high, the two regions are very likely to be part of the same material. On the other hand, a low value does not inform us much. Thus, it makes sense that a correct segmentation should maximize a combined certainty criterion drawn from the two measures at our disposal. To do this, we could maximize the following objective function:

$$E = \sum_{i\in G, j\in O} D_{ij} + \sum_{i\in G, j\in G} A_{ij} - \sum_{i\in G, j\in O} A_{ij} \qquad (1)$$

with respect to the region labels ("(G)lass" or "(O)ther"). The first term enforces large discrepancies between glass regions and non-glass regions. The second term enforces large affinities internal to the glass. The final term penalizes a segmentation for breaking any large affinities between the glass and non-glass. Unfortunately, maximizing this function is a combinatorial problem: In practice, even though most initial segmentations only leave a few hundred regions, having to enumerate an exponential number of combinations of these is still far too expensive.

We propose instead to change the original objective function definition and concentrate on the discrepancy between

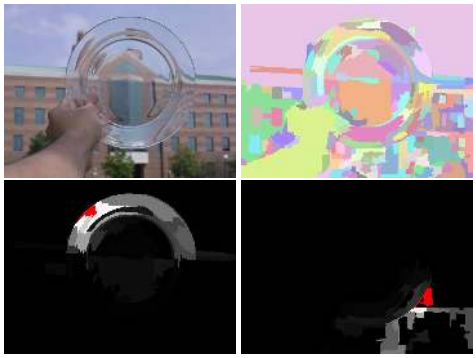

Figure 5. **Top row:** *an image with its initial segmentation based on color and texture.* **Bottom row:** *Two selected regions (red) and the affinity values of all other regions. Brighter intensities indicate higher values.*

adjacent regions. In this context, it is convenient to define

$$D_{xy} = b_{xy} D_{i_{xy}j}$$

to express the local discrepancy at the pixel level. Here i_{xy} is the region that contains point x, y and b_{xy} is an indicator variable whose value is 1 if point x, y is on the border of this region. The variable j is then the region on the opposite side of the border. Similarly, we define

$$A_{xy} = b_{xy} \left(\sum_{j\in G} A_{i_{xy}j} - \sum_{j\in O} A_{i_{xy}j} \right)$$

to express the pixels' affinity to the glass set.

Optimizing an objective function based on A_{xy} and D_{xy} over all possible pixel labels is still a combinatorial problem. Thus, we treat every pixel as a sample of an underlying continuous function and relax the original combinatorial problem into a continuous one. For this, we use the geodesic active contour framework of Casselles et al. [6], and define an objective function over the smooth boundary (*contour*) $\mathcal{C}_s$ of an evolving image region:

$$E(\mathcal{C}) = \int_0^1 g(B_{\mathcal{C}_s}) |\dot{\mathcal{C}}_s| ds,$$

where g is a monotonically decreasing function (e.g. Gaussian), and $|\dot{\mathcal{C}}_s|$ is a regularization term that tends to minimize the length of the contour and maximize its smoothness. For regions defined in the discrete image domain, $B_{\mathcal{C}_s}$ is replaced by a term B_{xy} (boundary strength) given in our case by:

$$B_{xy} = \alpha D_{xy} + (1-\alpha) A_{xy}$$

where α weights the significance of the discrepancy and affinity terms. We have defined D_{xy} and A_{xy} only at region borders since we want a subset of whole regions as our output.

We now wish to minimize this function. This can be done with a gradient descent method by differentiating over time [6]:

$$\frac{\partial}{\partial t}\mathcal{C} = g(B_{\mathcal{C}_s})\mathcal{K}\vec{\mathcal{N}} - (\nabla g(B_{\mathcal{C}_s}) \cdot \vec{\mathcal{N}})\vec{\mathcal{N}}$$

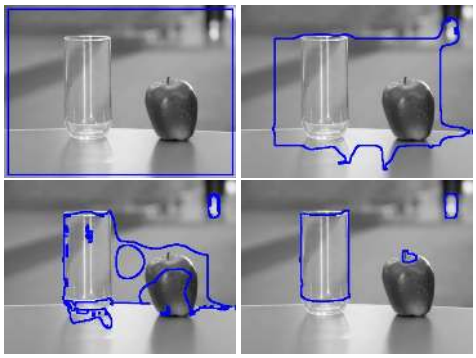

Figure 6. *Evolution of the level set (blue) as it segments the image into glass and non-glass. Note that two small regions have been classified incorrectly.*

where $\mathcal{K}$ is the Euclidean curvature with respect to the curve and $\vec{\mathcal{N}}$ the normal. This can be solved in a level set implementation by embedding this within a function ϕ whose 0 level set is the current contour. Given an initialization for ϕ we adjust it with the following evolution equation [6]:

$$\frac{\partial}{\partial t}\phi = g(Bc_s)\mathcal{K}|\nabla\phi| + \nabla g(Bc_s)\cdot\nabla\phi$$

In our implementation we initialize ϕ so that its 0 level set is near the border of the image and set α to 0.25 (Figure 6).

The evolution equation implicitly requires the curve to be smooth. We find this beneficial since trouble areas such as texture will usually have rough boundaries.

Note that by changing our objective function to fit the geodesic active contour framework, our final result may not be on the initial region boundaries (even though we restricted D_{xy} and A_{xy} to border pixels). This is wasted information, since we want a subset of these regions as our output. To account for this, we reset the contour every few iterations to match the boundary of the regions within it. This can be done by looking at the percentage of each region inside the current contour and removing all regions that have more than 50 % of its pixels outside.

We further enforce the smoothness of the boundary with a simple post-processing step where we examine each of the regions just outside the border of what has been labeled as glass. If adding this region reduces the overall perimeter of the object then it too is classified as glass. This is repeated until no further regions can be added.

4. Experiments and Results

We compare the proposed method to the ones proposed in [14] on their test data set of fifty images. Thirty five of the images contain glass objects in front of various backgrounds. Fifteen of the images contain no glass objects. In addition we train the glass classifiers used in D_{ij} on the same fifteen training images, six with glass objects in front of various backgrounds and nine with no glass at all. The results are shown in Table 1.

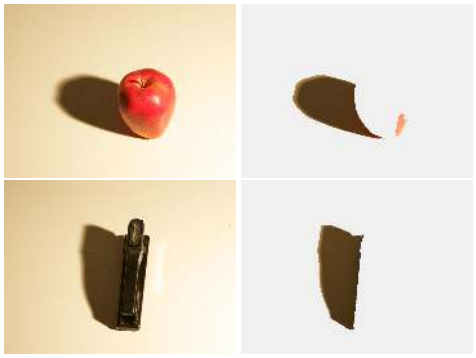

Figure 7. *A proof-of-concept illustration showing that the outlined technique can be generalized to find shadows with a discrepancy based on the appropriate cues. It is interesting to note that the stapler is actually darker than its shadow in the image.*

The proposed method obtains a precision (percentage of pixels labeled glass that are actually glass) of 77.03% which is higher than any of the edge based methods proposed in [14]. Some sample segmentations are given in Figure 8. Notice the two distinct glass objects in the third row, which could not be found with the edge based method. A couple of false positives can be seen in the second row, though most of the image is segmented correctly. In the first row there is a conspicuous piece missing out of the right side of the plate. Though its difficult to see in the small image one of the external building regions partially bleeds into the plate allowing for some of the glass regions to have a somewhat high affinity with this region. When this region is eventually removed it weakens those regions allowing the contour to pass by their boundaries. Similarly in the fourth row. In this case the orange region making up one of the wall planks crosses into the right bottle from time to time. Glass regions incorrectly linked to to this region are eventually weakened when the plank is removed. Of course if the initial segmentation merges some of the glass into the background we can not separate it later on. This can be seen in row two, where most of the base of the tea glass is merged into the table. Again in row five, the unmarked thin areas at the top of the bowl are part of the the blue region just above it. In row three an incorrect merging of part of the glass with the opening of the mug and the base of the bottle with part of the table results in false positives. In general, the initial segmentation does a good job of separating the region into uniform pieces of background and glass.

5. Conclusion

We have presented a region-based segmentation method for finding objects made of transparent materials such as glass. As argued by Adelson in [1], the visual vocabulary of materials includes much more than just texture. The recognition of transparency is one such example, but finding metallic, dull, or even greasy spots in an image would be interesting as well. Identifying a material's typical form, such as be-

Method	True Positive Rate	False Positive Rate	Precision
SVM on all cues	47.01%	3.09%	68.76%
Multiple SVM's + Weighted sum	83.94%	8.53%	58.78%
Multiple SVM's + Exponential model	88.30%	10.04%	56.04%
Multiple SVM's + Weighted sum (sampled)	79.72%	4.12%	73.70%
Proposed method	61.73%	2.67%	77.03%

Table 1. *Results of the various classifiers described in [14] all tested on a test set of 50 images where glass pixels were marked by hand. True positive rate, or recall indicates the percentage of glass pixels that were correctly identified. The false positive rate indicates the percentage of non-glass pixels that were identified as glass. Precision is the percentage of identified glass pixels that were actually glass.* **From top to bottom:** *an SVM on all six values of the cues, a classifier consisting of a weighted sum of the sub-classifiers described in section 2.1, a classifier consisting of the outputs of the sub-classifiers combined through an exponential model, a classifier consisting of a weighted sum of the sub-classifiers and trained on only a sample of the training data [14], and the proposed segmentation method.*

ing granular, fibrous, porous, nodular, columnar, foliated, or scaly is interesting as well [1]. While characterizing a material's luster as metallic or mirror-like may be difficult when looking at a single image region, it might be accomplished using a binary measure as we have here. In the case of mirror-like surfaces, we might also compare two regions to see if one could be a reflected version of the other (possibly with some distortion).

We would like to point out that the general approach that we have described here can also be applied to the task of finding shadows with different, appropriate cues to define the discrepancy measure (Figure 7). Color cues, such as as the ratios of their components [3] have been used to remove shadows [13]. In addition it has been observed that some color spaces tend to have channels that are invariant to shadows [20]. This property lends itself to another shadow cue based on color. Of course, texture should be a cue since a shadow should not modify an objects texture, thus the texture would have to be very similar in two regions if one is to be considered a shadow-covered version of the other. We intend to explore this further in future work.

Acknowledgements. This work is partially supported by the National Science Foundation under grants IIS 03-08087 and IIS 05-35152, Toyota Motor Corporation, and the Beckman Institute.

References

[1] E. Adelson. On seeing stuff: The perception of materials by humans and machines. *SPIE*, pages 1–12, 2001.

[2] E. Adelson and P. Aniandan. Oridanl characteristics of transparency. *AAAI*, pages 77–81, 1990.

[3] K. Barnard and G. Finlayson. Shadow identification using colour ratios. *8th Color Imaging Conference*, 2000.

[4] M. Ben-Ezra and S. Nayar. What does motion reveal about transparency? *ICCV*, pages 1025–1032, 2003.

[5] B. Caputo, E. Hayman, and P. Mallikarjuna. Class-specific material categorisation. *ICCV*, 2005.

[6] V. Casselles, R. Kimmel, and G. Sapiro. Geodesic active contours. *ICCV*, 1995.

[7] P. Felzenszwalb and D. Huttenlocher. Efficient graph-based image segmentation. *IJCV*, pages 167–181, 2004.

[8] D. Forsyth and J. Ponce. *Computer Vision, A Modern Approach*. Prentice Hall, 2003.

[9] S. Hata, Y. Saitoh, S. Kumamura, and K. Kaida. Shape extraction of transparent object using genetic algorithm. *ICPR*, pages 684–688, 1996.

[10] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. *IJCV*, 1:321–331, 1987.

[11] S. Lazebnik, C. Schmid, and J. Ponce. Affine-invariant local descriptors and neighborhood statistics for texture recognition. *ICCV*, 2003.

[12] A. Levin, A. Zomet, and Y. Weiss. Seperating reflections from a single image using local features. *CVPR*, 2004.

[13] M. Levin and J. Bhattacharyya. Removing shadows. *Pattern Recognition Letters*, pages 251–265, 2005.

[14] K. McHenry, J. Ponce, and D. A. Forsyth. Finding glass. *CVPR*, 2005.

[15] D. Miyazaki, M. Kagesawa, and K. Ikeuchi. Polarization-based transparent surface modeling from two views. *ICCV*, pages 1381–1386, 2003.

[16] H. Murase. Surface shape reconstruction of an undulating transparent object. *ICCV*, pages 313–317, 1990.

[17] P. Nillius and J. Eklundh. Classifying materials from their reflectance properties. *ECCV*, pages 366–376, 2004.

[18] M. Osadchy, D. Jacobs, and R. Ramamoorthi. Using specularities for recognition. *ICCV*, pages 1512–1519, 2003.

[19] J. Platt. *Advances in Large Margin Classifiers*, chapter Probabilistic Outputs for Support Vector Machines and Comparisons to Regularized Likelihood Methods, pages 61–74. MIT Press, 2000.

[20] E. Salvador, A. Cavallaro, and T. Ebrahimi. Shadow identification and classification using invariant color models. *ICASSP*, pages 1545–1548, 2001.

[21] M. Singh and X. Huang. Computing layered surface representations: An algoritm for detecting and seperating transparent overlays. *CVPR*, 2:11–18, 2003.

[22] J. Solem, H. Aanaes, and A. Heyden. Pde based shape from specularities. *LNCS*, pages 401–415, 2003.

[23] R. Szeliski, S. Avidan, and P. Aniandan. Layer extraction from multiple images containing reflections and transparency. *CVPR*, 2000.

[24] M. Varma and A. Zisserman. Classifying images of materials: Achieving viewpoint and illumination independence. *ECCV*, pages 255–271, 2002.

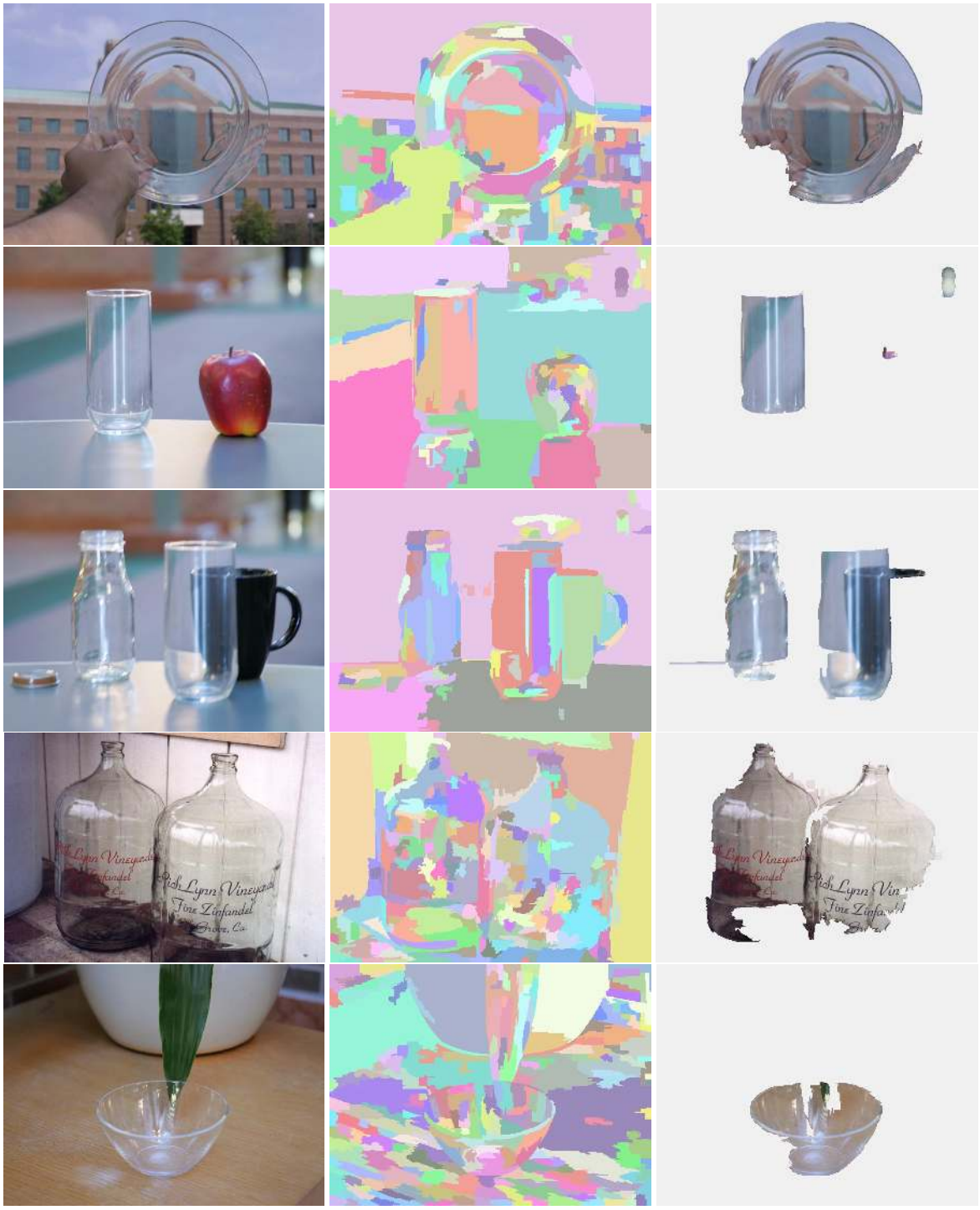

Figure 8. *Test images and output from the described method.* **Left:** *Five sample test images,* **Middle:** *The initial segmentation based on color and texture,* **Right:** *Regions labeled glass.*