Volume 0 (1981), Number 0 pp. 1–2 COMPUTER GRAPHICS *forum*

Sketching in Gestalt Space: Interactive Shape Abstraction through Perceptual Reasoning

J. Kratt[1], T. Niese[1], R. Hu[2], H. Huang[2], S. Pirk[3], A. Sharf[4], D. Cohen-Or[5], O. Deussen[1]

[1]University of Konstanz, Germany, [2]Shenzhen University, China, [3]Stanford University, USA,
[4]Ben-Gurion University of the Negev, Israel, [5]Tel Aviv University, Israel

1. Conjoining Gestalt Groups

In this Section we provide a detailed overview about the computation of the different parts of the energy function that is used to resolve conflicts between Gestalt groups in 3D. In our work the input consists of 3D models that have already been segmented into low-level elements. In order to find all potential Gestalt groups formed by these elements, we extend the 2D Gestalt rules to 3D. Similar to Nan et al. [NSX*11], we build a proximity graph G that connects the 3D elements of our scene to their direct neighbors. For each element p_i one node in the graph is constructed. Furthermore, for each element p_i, we find its closest neighbors, p_j, and connect them with an edge e_{ij} having an associated weight that is related to the Hausdorff-distance between the elements. Based on this graph we find potential Gestalt groups.

Once we have identified Gestalt groups, each element p is assigned labels f_p of the corresponding Gestalt groups they belong to. Some elements get more than one label, resulting in an over-segmented scene. Nan et al. employ an approximate multi-label graph-cut energy minimization [DOIB12] to solve this combinatorial problem efficiently. The result is the joint labeling f, which assigns a single Gestalt label to each element and minimizes an objective energy function. This function consists of three terms: data, smoothness and label costs. In this Section we explain our adaptations to the energy function in order to work on the 3D elements of our model. In the following L_P, L_S and L_R are the sets of elements that belong to a proximity, similarity or a regularity group.

1.1. Label Cost

Within the optimization, the label cost term favors configurations with only few and cheap labels and is defined as:

$$F_{cost} = \sum_{l \in L} h_l \cdot \delta_l(f),$$

where L describes the entire set of labels and h_l is the actual cost value of label l. $\delta_l(f)$ being an indicator function. This function has either a value of 1, if there exists at least one element that is assigned the label l, 0 otherwise. The label cost given by h_l depends on the type of Gestalt group and measures the affinity for a specific principle. Since our implementation works on 3D elements we have to adjust the individual costs compared to the definition given by Nan et al. In particular, for proximity groups we measure the volume density in 3D space. For regularity groups we directly exploit the structure of the pattern, defined by the underlying path in the proximity graph, rather than computing the cost in frequency space.

- *Proximity Groups:* The cost value for a group that follows the law of proximity is defined by the difference between the volume of the convex shape and the unified volume of the elements : $h_l =_{p \in L_P} vol(CH(p)) - \bigcup(vol(p))$.
- *Similarity Groups:* Similarity label cost is computed as the similarity variance among elements within the group: $h_l =_{(p_i,p_j) \in L_S} var(sim(p_i,p_j))$, where $sim() \in [0,1]$ measures the similarity between two elements. $sim()$ returns a value of 0 if two shapes are equal.
- *Regularity Groups:* For regularity Gestalts, the cost is defined by the inverse density multiplied by the variance of edge lengths and angles between edges of the corresponding path: $h_l =_{p \in L_R} (vol(CH(p)) - \bigcup(vol(p))) \times var(||e_i||) \times var(\alpha_j) \times ||L_R||^{-1}$, with $0 \leq i \leq n$ and $0 \leq j < n$ for a given path $(e_0, e_1, \ldots, e_n)$ in G that was detected to follow a regular pattern.

1.2. Smoothness Cost

Elements that are close together are more likely to belong to the same Gestalt group. To consider this in the optimization, Nan et al. define the smoothness cost as the inverse Hausdorff-distances between neighboring elements, which can also be directly applied to our 3D elements:

$$F_{smooth} = \sum_{p,q \in N} = d(p,q)^{-1},$$

where N is the set of all neighbor elements in G and $d(p,q)^{-1}$ represents the inverse Hausdorff-distances between elements p and q. The inverse distances are further normalized into the range $[0,1]$ using the minimal and maximal distance appear in the graph.

1.3. Data Cost

The data cost term reflects how well elements fit to their assigned Gestalt groups and is defined as:

$$F_{data} = \sum_{p \in P} D(p, f_p),$$

where $D(p, f_p)$ is the data cost value for an element p assigned to label f_p. The evaluation of the individual cost depends on the type of Gestalt group. Similar to the label cost, we have to adapt the computations to work on 3D elements:

- *Proximity Group:* The data cost for an element p with respect to its proximity group L_P is measured as the closest Hausdorff-distance of p from all other elements in the group: $D(p, f_p) = \min_{q \in L_P \setminus \{p\}} d(p, q)$.
- *Similarity Group:* The data cost for an element p that belongs to a similarity group L_S is defined by the average similarity distance of p from all other elements in the group: $D(p, f_p) = \frac{1}{|L_S \setminus \{p\}|} \sum_{q \in L_S \setminus \{p\}} sim(p, q)$. Again, $sim() \in [0, 1]$ is the used similarity measure, returning 0 if two shapes are equal.
- *Regularity Group:* The data cost of an element p that is part of a set L_R of regularly aligned elements is computed by measuring the distance of p to its ideal element p', that would be perfectly aligned with the regular pattern: $D(p, f_p) = d(p, p')$.

In order to compute the data cost for an element that belongs to a regularity group, we have to determine the ideal position of the element, that perfectly aligns with the regular structure of the group. This is done by taking the average edge length and the average angle of the underlying path. Having these two values, we can construct a path that has no variation in edge length and angles between two successive edges. This path serves as approximation for the perfect regular aligned pattern and we can compute the distances.

2. Energy Function

To account for the visual importance of Gestalt groups we finally introduced our visibility terms (group dominance and element visibility) into the energy function:

$$E(f) = \sum_{p \in P} (1 - A(p, f_p)) \cdot D(p, f_p) \sum_{p,q \in N} V_{p,q} + \sum_{l \in L} (1 - \bar{\tau}_l) \cdot h_l \cdot \delta_l(f), \quad (1)$$

where $A(p, f_p)$ is the visibility value and $D(p, f_p)$ the data cost for an element p if the label f_p is assigned to it. $V_{p,q}$ is the smoothness cost for two neighboring elements p and q. The term $h_l \cdot \delta_l(f)$, weighted by the average group dominance $\bar{\tau}_l$, represents the label cost with L being the entire set of labels. Since the optimization seeks for minimizing the energy function, we have to invert the visibility terms in order to favor visible groups for abstraction.

References

[DOIB12] DELONG A., OSOKIN A., ISACK H. N., BOYKOV Y.: Fast approximate energy minimization with label costs. *Int. J. Comput. Vision 96*, 1 (Jan. 2012), 1–27. 1

[NSX*11] NAN L., SHARF A., XIE K., WONG T.-T., DEUSSEN O., COHEN-OR D., CHEN B.: Conjoining gestalt rules for abstraction of architectural drawings. *ACM Trans. Graph. 30*, 6 (Dec. 2011), 185:1–185:10. 1