1 Introduction

In this paper we proposed an algorithm for blending between two given key polygonal shapes. Generally speaking, 2-D shape blending requires two mainstays: one is vertices correspondence and the other is shape interpolation. Some approaches use matrix interpolation which sometimes causes shrinkages and kinks because a transformation common to all vertices may not exist. Other methods try to interpolate the length of edges and the angles formed by two adjacent edges[1]. This article makes an attempt based on center of mass to generate in-betweens more fast.

2 Center of Mass of Polygonal Shape

Two key polygons should have same number of vertices after vertices correspondence has been established. Given two polygonal shapes: source shape $P^s$ and target shape $P^t$ with the same number of vertices.

$$P^s = \{P^s_0, P^s_1, \ldots, P^s_n\} \quad P^t = \{P^t_0, P^t_1, \ldots, P^t_n\}$$ (1)

For each corresponding vertices, the interpolation is proceeded by:

$$P(t) = (1-t)P^s + tP^t$$ (2)

where $t \in (0, 1)$ can be regarded as time.

For each key polygonal shapes, given the coordinates $(x_i^s, y_i^s)$ of vertices $P^s_i$ of shape $P^s$ and the coordinates $(x_i^t, y_i^t)$ $(i = 0, 1, \ldots, n)$. We can obtain the center of mass $M^s(M^s_x, M^s_y)$ and $M^t(M^t_x, M^t_y)$ for each polygonal shape, respectively.

$$M^s_x = \frac{1}{n+1} \sum_{i=0}^{n} x_i^s; \quad M^s_y = \frac{1}{n+1} \sum_{i=0}^{n} y_i^s$$ (3)

$$M^t_x = \frac{1}{n+1} \sum_{i=0}^{n} x_i^t; \quad M^t_y = \frac{1}{n+1} \sum_{i=0}^{n} y_i^t$$ (4)

![Fig.1 distances and angles variables](image)

We begin by obtaining the definitions of $P^s$ and $P^t$ by calculating distances $d_i^s, d_i^t$ from center of mass to vertices. And we also can compute the directed angles of vector $M^sP^s_i$ from z-axis and $M^tP^t_i$ shown in Figure 1. Here for the sake of convenience, We assume that counter-clockwise angles are positive as convention. The distances and angles are

$$d_i^s = |M^s - P^s_i| \quad d_i^t = |M^t - P^t_i|$$

$$\theta_i^s, \quad \theta_i^t (i = 0, 1, \ldots, n)$$ (5) (6)

3 Interpolating Center of Mass and Angles

After established vertices correspondence, using the definition of 2-D polygonal shape mentioned in section 2, the intermediate polygon shape can be calculated by interpolating center of mass, distances from center of mass to vertex and vector angles. Here $i = 0, 1, \ldots, n$.

$$M_x = (1 - t) M^s_x + t M^t_x \quad M_y = (1 - t) M^s_y + t M^t_y$$ (7)

$$d_i = (1 - t) d_i^s + t d_i^t \quad \theta_i = (1 - t) \theta_i^s + t \theta_i^t$$ (8)

4 Experimental Results

Fig.2 and Fig.3 shows experimental results of interpolation based on center of mass. Although interpolation of center of mass is linear, vertices interpolation path is nonlinear. Experience suggests that it work well enough and more smooth for most of realistic cases.

![Fig.2 In-betweening](image)  ![Fig.3 In-betweening](image)

between two line-draws between two line-draws

5 Conclusion

A new method was proposed for interpolation between two given polygonal shapes. The source polygon and target polygon was defined by center of mass and vectors from center of mass to vertices. The center of mass, distance from center to vertex and angles of vectors were interpolated for generating the in-between polygons.

参考文献