Particle Analysis by 3-D Modeling

Mark Vickers, Hong Lu, Brian Spears, Justin Scott, Matt Hall, Larry Baxter

Chemical Engineering Department, Brigham Young University, Provo, UT

Objectives

We are working on an algorithm that will read in three orthogonal images of any type of particle and calculate the surface area, volume, and produce a 3D, interactive model with a particle surface temperature map.

Applications

Some immediate applications will be for the study of biomass particle going through a entrained flow reactor and black liquor burning in TG/MS. This algorithm will also be useful in our study of particles such as popcorn ash.

Approach

Image Processing

For each 2D particle image, the perimeter of the particle is found by first reading in the pixel intensities and then running that data through an algorithm that walks around the particle guided by the pixel intensity values.

Image Alignment

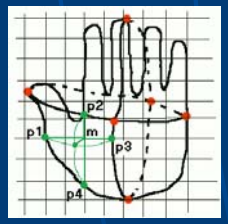
In case of size distortions due to a slight difference in the distance from the particle to each camera, all images are proportioned to the same size. Cubic spline interpolation method is then used to map each image onto the other, providing the basic three wire frames that will be used to get the surface. This can be done given that we know the extreme points in each direction.

Point Interpolation

We are currently using Inverse Distance Weighting (IDW) as our method of interpolating the surface points, where the points are calculated from a weighted average of the neighboring points. This method is often used in geostatistics and spatial statistics. There is one major limitation to using this method, that is, it does not handle concavities on the XZ plane image or the ZY plane image. We are currently working to find ways to deal with these concavities. As we have approached this problem, we have thought of the hand as a good example of concavity, and something we should be able to model. The IDW formula we use is illustrated below with the aid of an image of a hand.

$$d_p = \sqrt{x^2 + y^2}$$

$$z_m = \frac{\sum_1^N z_p * d_p^{-n}}{\sum_1^N d_p^{-n}}$$



Surface Area and Volume

From the interpolated surface points, we run an algorithm that groups those points into tetrahedrals. We take the sum of the volumes of each tetrahedral to be the particle volume and we keep track of which tetrahedral faces are on the hull so that we can get the total surface area by summing their individual surface areas.

Preliminary Results

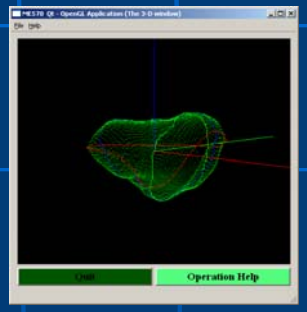
Input: These input images are taken from the popcorn deposit particle that is attached to this poster:

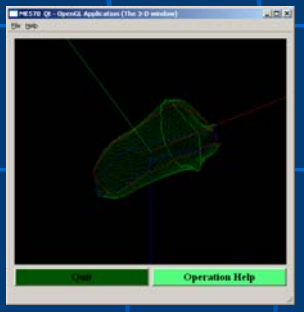

XZ plane

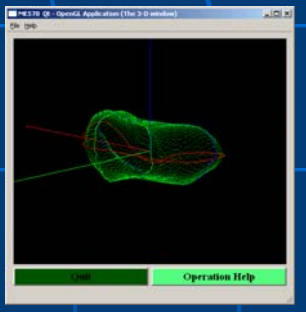

YZ plane


XY plane

Output:

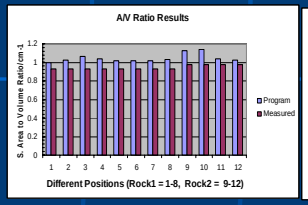


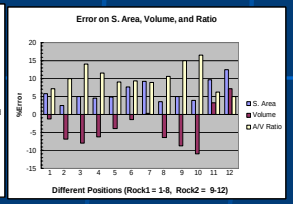




Particle Surface Area: 1.58686e-004 (m^2) Particle Volume: 1.08726e-007 (m^3)

Error analysis





Volume	
Avg. Error:	5.06 %
Mean Error:	5.04 %

Ratio	
Avg. Error:	10.29 %
Mean Error:	9.675 %

Surface Area	
Avg. Error:	6.19 %
Mean Error:	4.96 %

Future Work

- The temperature map is currently being implemented. The temperatures will be mapped onto the 3D model according to the red, green, and blue pixel intensities of the images.
- We would like to generate this model from any number of images to as low as 2, and we want to allow them to be taken from angles that are not to be assumed as orthogonal.
- When creating tetrahedrals out of the interpolated points, concavity presents a challenge. When modeling a hand, for example, the tetrahedralization algorithm does not know where the edge of one finger begins and the other starts, This results in a volume that would include the space between the fingers as well as an inaccurate surface area and surface rendering. We will try to face this challenge with the idea that every object with concavity can be broken down into smaller convex objects,

Acknowledgement

US DOE/EE Office of Industrial Technologies & Sandia National Laboratories