Application of Morphological Erosion to a Lightweight Control Arm Design

Introduction

Lightweight structure design in automotive and aviation engineering focuses on reducing weight while maintaining strength and performance. Components like suspension control arms are often optimized for minimal mass using techniques such as topology optimization. In this context, mathematical morphology, a set of image processing operations that modify shapes, can be a useful tool for refining designs and simulating manufacturing effects. Morphological erosion, in particular, is an operation that shrinks the foreground region of a binary image. In image analysis, basic morphological operators like erosion and dilation are used to probe and alter shapes; erosion essentially removes pixels on object boundaries, causing objects to become thinner and smaller. By applying erosion to a binary mask of a component, we can simulate the removal of material (making the part uniformly thinner) and eliminate small-scale features. This has practical significance in design: for instance, ensuring that a structure remains viable even if manufactured slightly undersized (simulating a machining tolerance or material etching error). In the following sections, we describe how a control arm image was converted to a binary mask, discuss the erosion procedure with various kernel sizes, and analyze the outcomes both visually and in performance terms.

Binary Mask Conversion Methodology

The first step is to obtain a binary mask of the control arm from a source image. The control arm in this study (an automotive suspension lower control arm) was provided as a colored image. We converted this image to a binary format where the foreground (the metal arm) is white (pixel value 1) and the background is black (pixel value 0). To achieve this, the image was converted to grayscale and a high threshold was applied so that the white background (and any specular highlights) become 0 while all darker pixels corresponding to the part become 1. We also performed minor cleanup to ensure that internal edges and outlines in the image are filled in the mask. The result is a binary silhouette of the control arm, including its outer contour and any cut-out holes. This binary representation is crucial because morphological operations are typically applied to binary images representing shape geometry. It provides a simplified model of the component's shape, suitable for morphological processing.

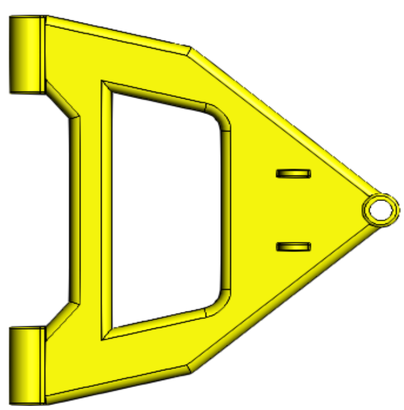

Figure.1 Example of Colored 3D CAD model of a lower control arm

Morphological Erosion Process with 3×3, 5×5, 7×7 Kernels

Morphological erosion was applied to the control arm's binary mask using three different square kernels (structuring elements) of sizes 3×3, 5×5, and 7×7. In all cases, the operation scans the binary image and erodes the white regions: a given foreground pixel remains white only if all pixels in its neighborhood (defined by the kernel size) are also white; otherwise, it is set to black. Equivalently, erosion can be viewed as "shrinking" the shape by one pixel layer for a 3×3 kernel (since the 3×3 kernel extends 1 pixel in each direction from a center), by two layers for a 5×5 kernel, and by three layers for a 7×7 kernel (generally, an $n \times n$ square kernel removes a border approximately $[\frac{n}{2}]$ pixels thick from the object). We expect that using larger kernels will more aggressively erode the object: small protrusions or thin connections will disappear and open holes will expand in size. Below, we present the visual results of erosion on the control arm mask for each kernel size, along with descriptions of the observed changes.

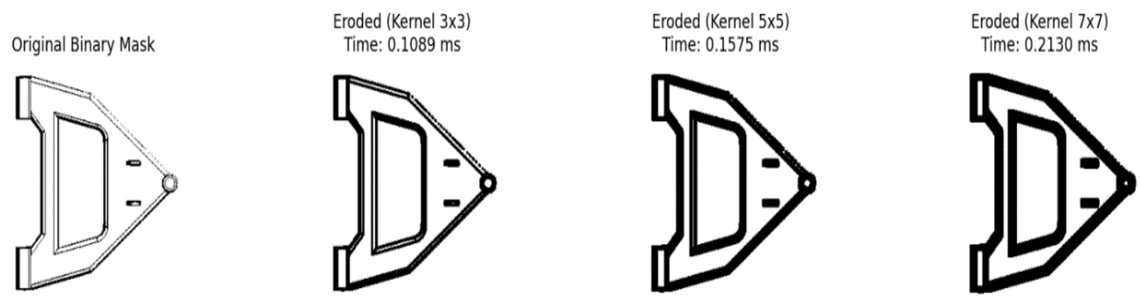

Figure 2. Morphological Erosion Results on Control Arm Image Using Different Kernel Sizes

Performance Analysis of Erosion Operations

We evaluated the computational performance of the morphological erosion operation applied to a binary mask of an automotive control arm using three square kernels: 3×3, 5×5, and 7×7. All operations were performed on an image of approximately 400×400 pixels.

Morphological erosion involves examining each pixel's neighborhood as defined by the kernel size. A larger kernel requires more computations per pixel because more neighboring pixels must be evaluated. For example, a 3×3 kernel examines 9 neighboring pixels per location, a 5×5 examines 25, and a 7×7 examines 49. This increase in kernel size leads to more intensive operations per pixel and, typically, longer processing times.

In this experiment, execution time was measured by averaging 100 repeated erosion operations using high-resolution timers. The results clearly show that execution time increases as kernel size increases:

Table: Measured Execution Times for Erosion Operations

Kernel size	Pixels in kernel	Execution time (ms)
3 x 3	9	0.108
5 x 5	25	0.157
7 x 7	49	0.213

This monotonic increase reflects the expected behavior of the erosion algorithm: more neighborhood comparisons result in longer per-frame processing time. The differences, however, remain small, all operations were completed in under 0.25 milliseconds, which highlights the efficiency of morphological operations on modern hardware, especially for images of moderate size and low complexity.