



# CUDA-based Rendering of 3D Minkowski Sums

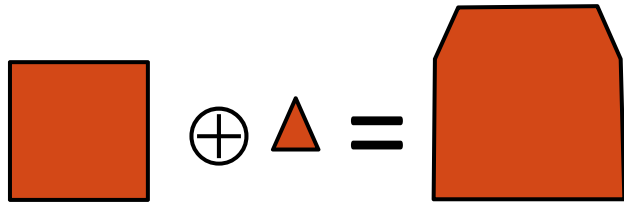
CS267 Project

Wei Li

# Background and Motivation

- Minkowski sum

- $C = A \oplus B = \{a + b \mid a \in A, b \in B\}$



- 3D Minkowski sum

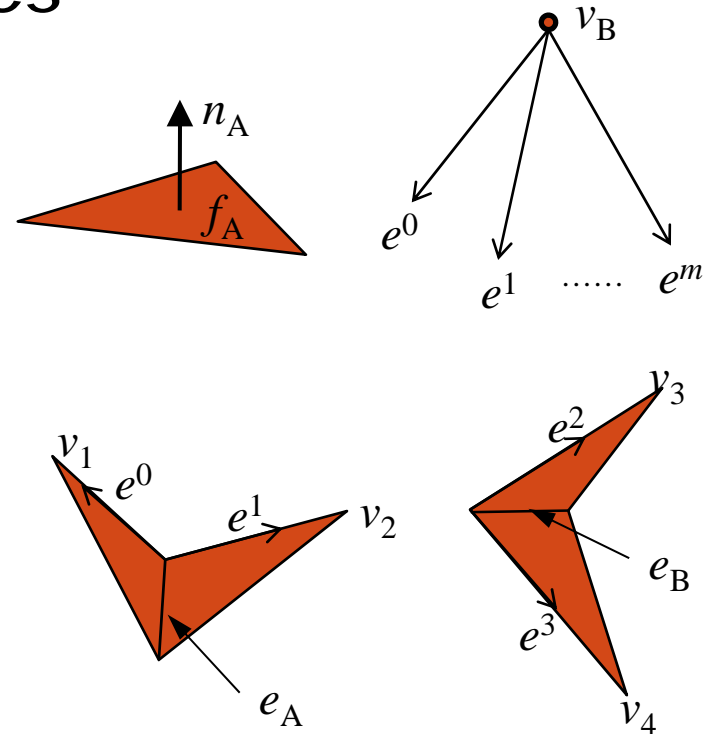
- $O(m^3n^3)$  complexity for concave objects
  - Existing computation algorithms very slow
    - Several minutes for simple objects with hundreds of triangles

- Motivation

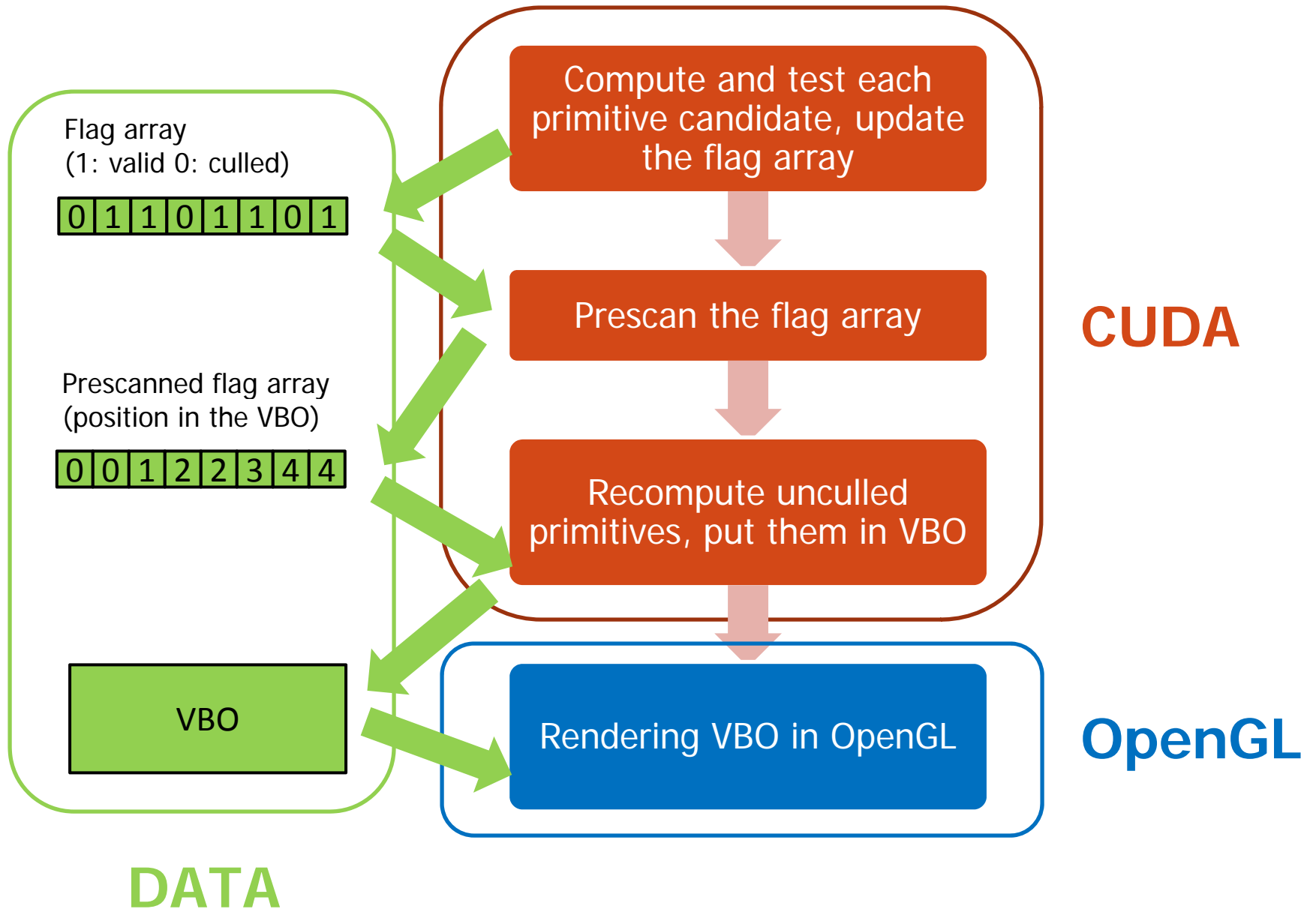
- Fast visual feedback of Minkowski sum
    - Input as triangulated polyhedra
  - Fast morphing
    - $tA \oplus (1-t)B$

# Surface Primitives

- Surface primitives of  $\partial C$ 
  - Triangles ( $f_A \oplus v_B$ , or  $f_B \oplus v_A$ )
  - Quads ( $e_A \oplus e_B$ )
  - May be trimmed
- Culling of interior primitives
  - $f_A \oplus v_B$  valid if
    - $\forall e^i, n_A \cdot e^i \leq 0$
  - $e_A \oplus e_B$  valid if
    - $\forall e^i, (e_A \times e_B) \cdot e^i$  constant sign
    - Not valid if  $e_A$  or  $e_B$  concave

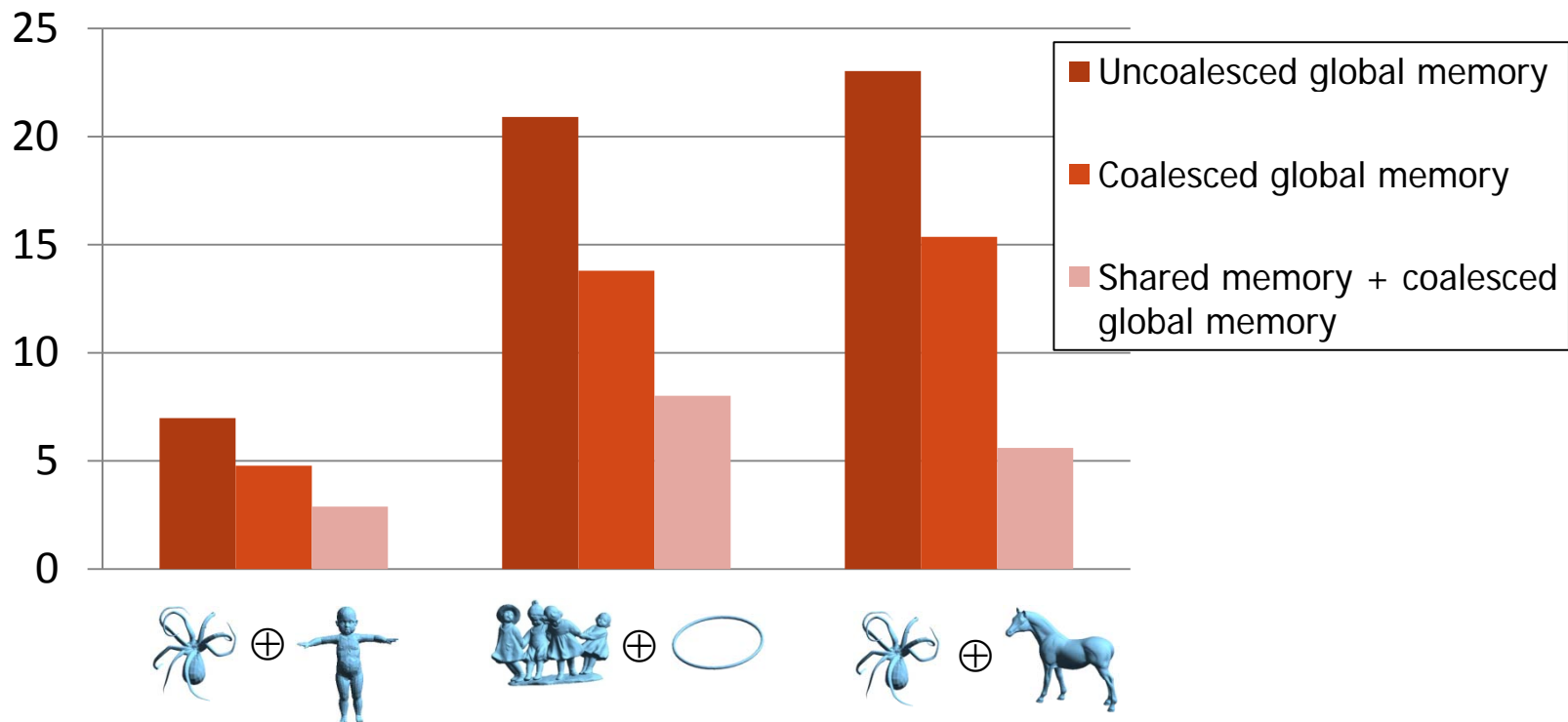


# Algorithm Overview



# Implementation on CUDA

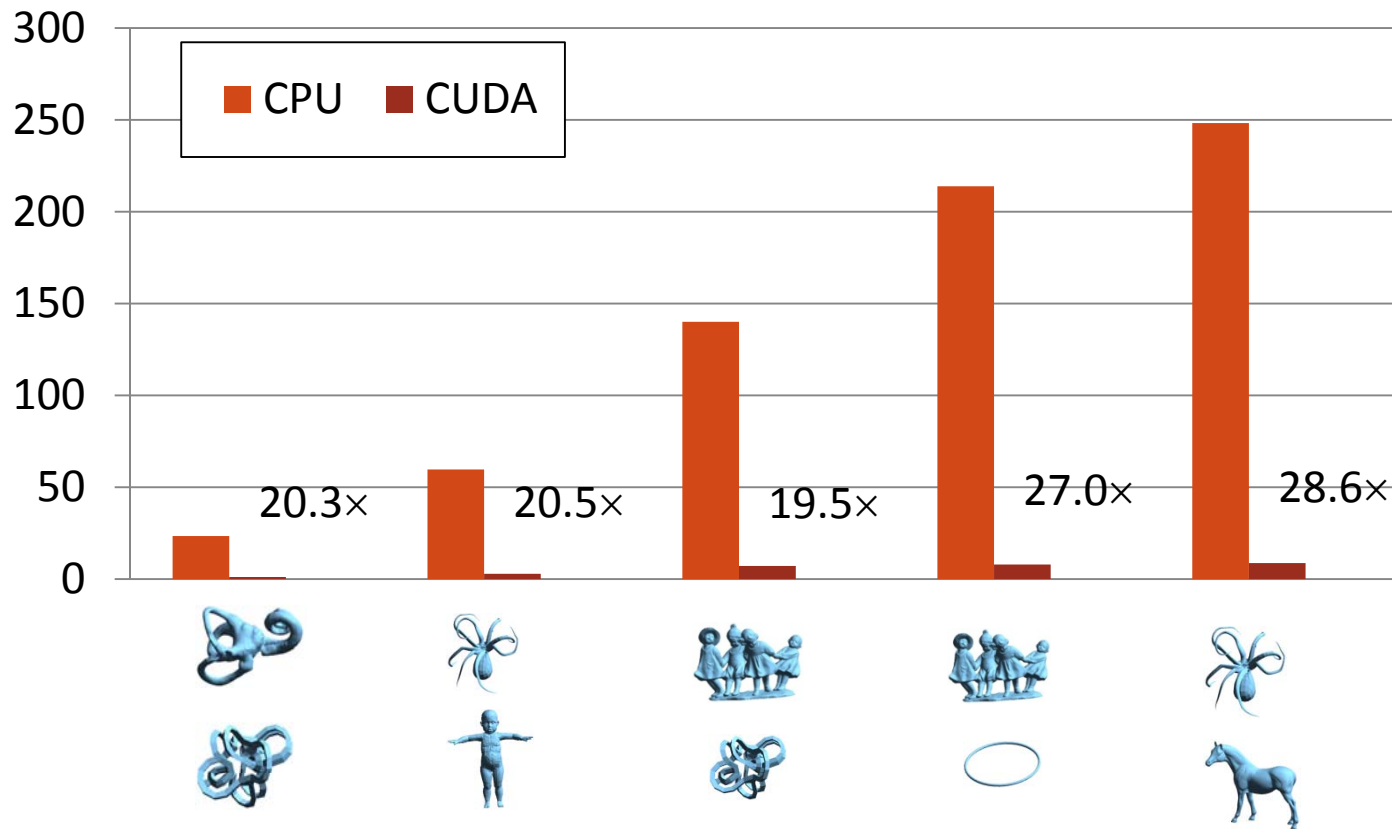
- Block size:  $16 \times 16$ 
  - #triangles, #vertices, #edges rounded up by 16
  - Each thread compute a primitive
- Memory Optimization



**Culling time with different memory techniques**  
(second)

# Timing Results

## Time for Rendering Minkowski Sums (second)



CPU: 3.00 GHz CPU w/ 2 GB RAM  
GPU: NVIDIA GeForce 9600 GT

# Rendering Results

