CFU: Multi-Purpose Configurable Filtering Unit for Mobile Multimedia Applications on Graphics Hardware

Chih-Hao Sun, Ka-Hang Lok, You-Ming Tsao,
Chia-Ming Chang, Shao-Yi Chien

Media IC & System Libratory
National Taiwan University

Outline

- Motivation
- Configure filtering unit (CFU)
- Configurability in CFU
- Related hardware and cache design
- Evaluations results
- Conclusions
- Future works

Mobile Multimedia Platform

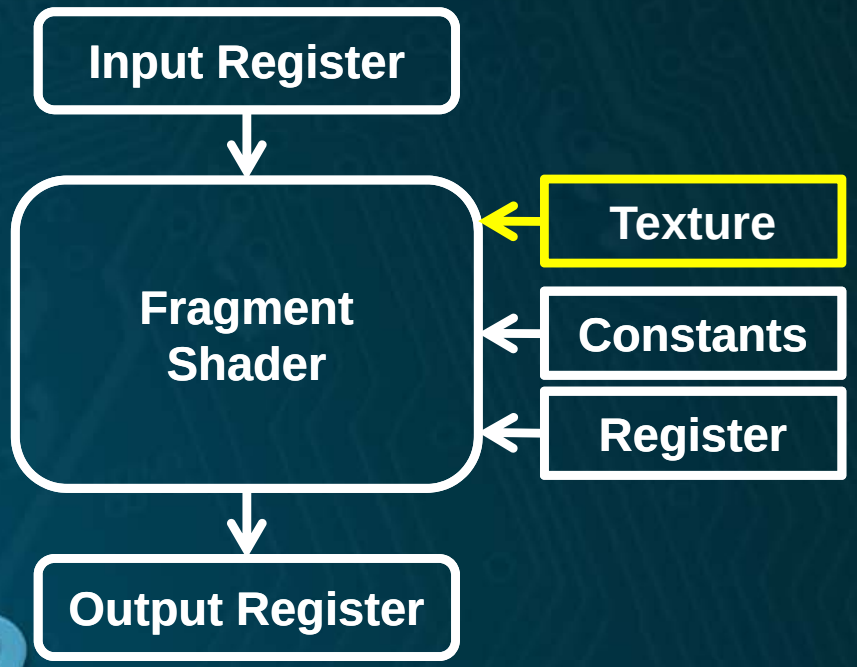
- The newest mobile multimedia platforms (phones) can support multiple functions.
  - Offer almost all advanced multimedia features.

Multi-Purpose Multimedia Subsystem on GPUs

- Mobile GPUs are designed with highly parallel stream processing architecture for multimedia applications.
  - We can establish a multi-purpose mobile multimedia subsystem on mobile GPUs.

Texturing in GPGPU

- The texture-based stream processing model is adopted.
  - Reference data is stored in the texture buffer and accessed though the texture unit.

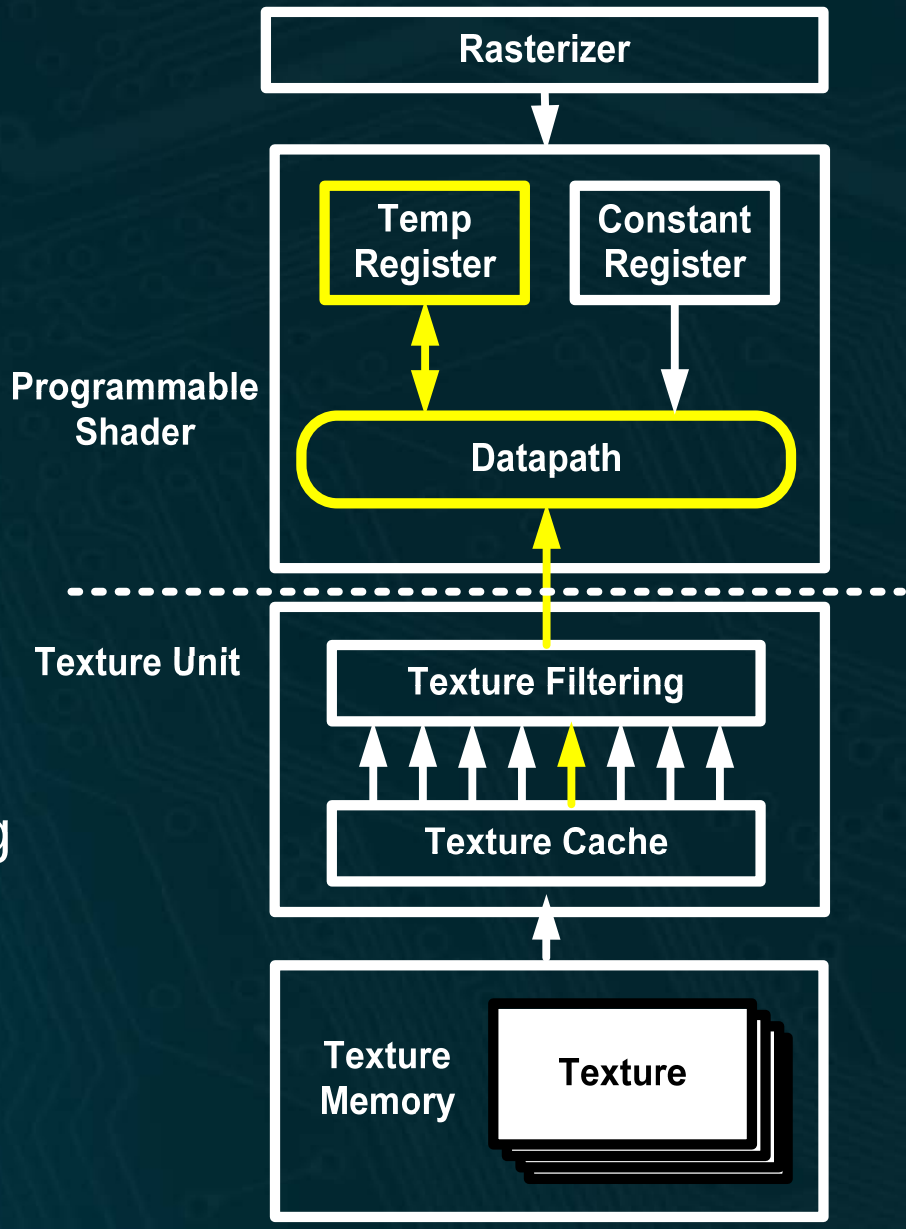


Reference Frame: ME , MC
Reference Coefficient: DCT
Input Frame: Edge enhancement

ME: Motion Estimation
MC: Motion Compensation
DCT: Discrete Cosine Transform

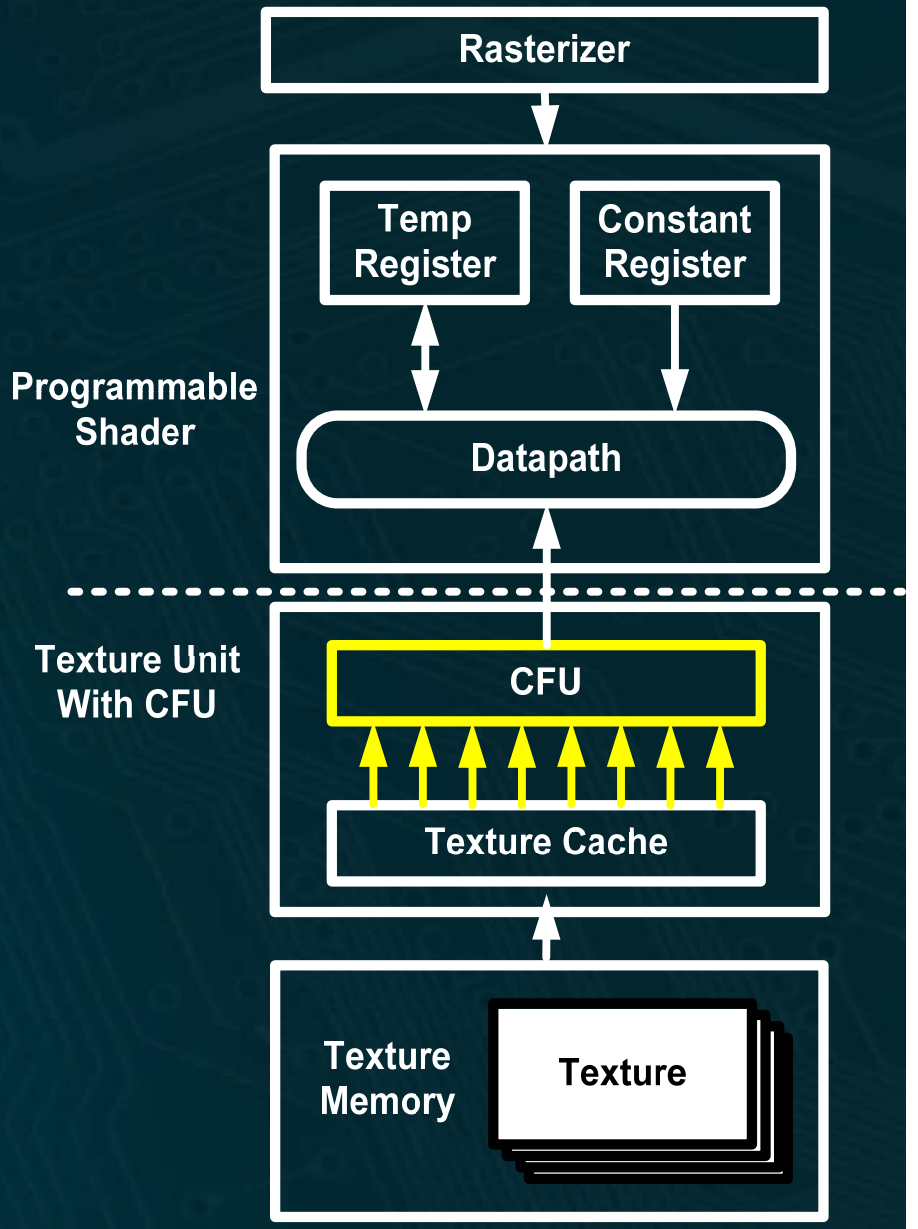
Traditional Architecture

- Programmable Shader
  - Limited register number and access ability.
  - Limited bandwidth between the shader and texture unit.
- Texture unit
  - High bandwidth between the filtering unit and texture cache.
  - Limited capability.



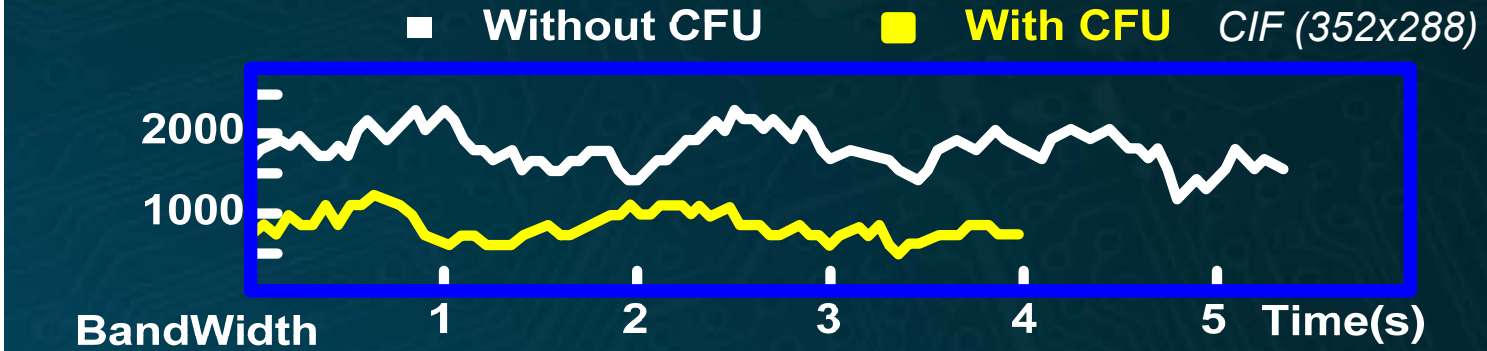
With Configurable Filtering Unit

- Relieve the loading of programmable shader.
  - Reduce register usage.
  - Fewer texture loading instructions.
- Increase the utilization of texture unit.
  - Execute more computations in the filtering unit.
  - Maximum the bandwidth usage between the filtering unit and cache.

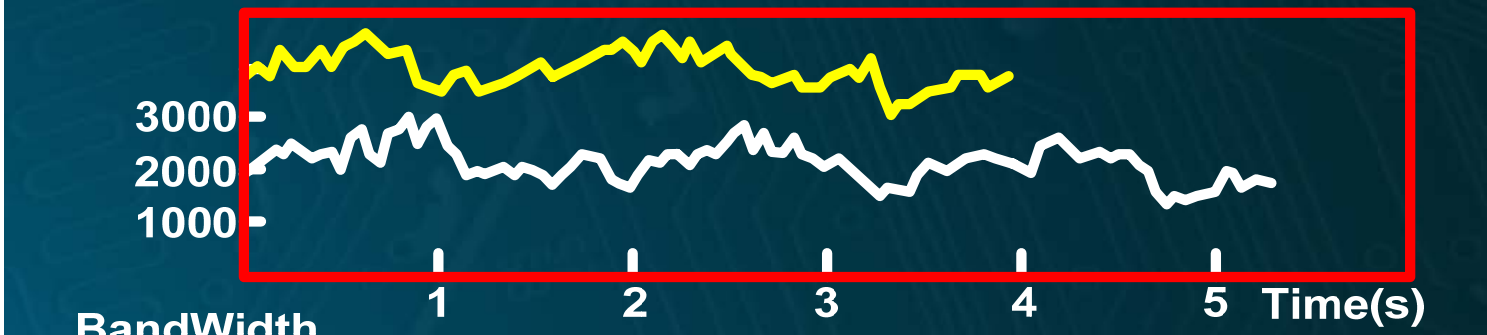


Effect of CFU

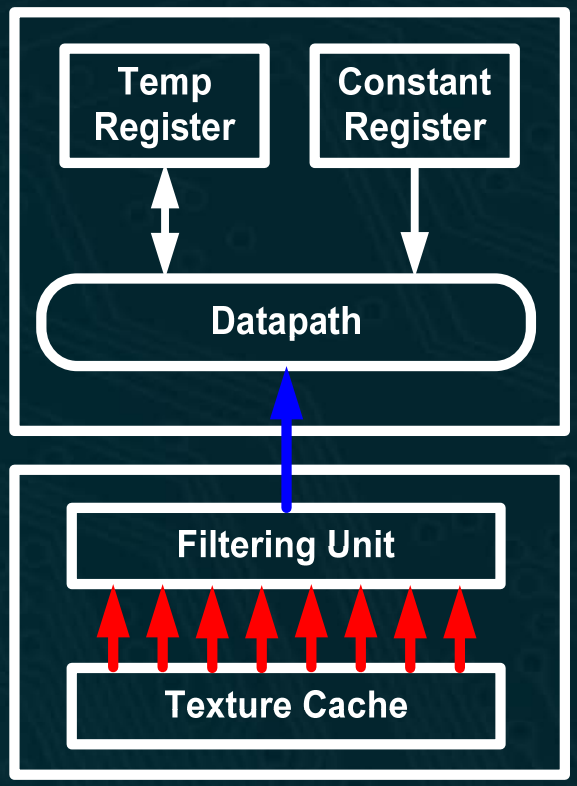
- Take H.264/AVC motion compensation as an evaluation case:



Bandwidth between shaders and texture unit

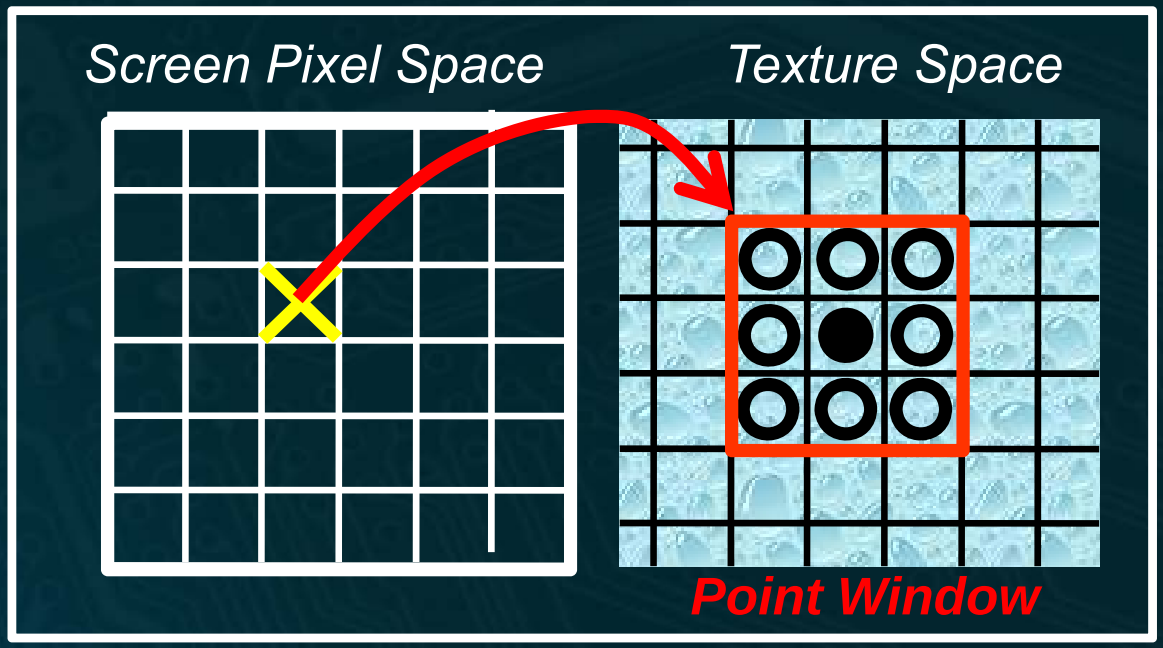


Bandwidth between filtering unit and texture cache



Configurability in CFU(I)

- Various point windows

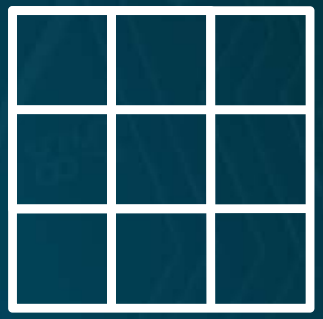


Bilinear Filtering

Trilinear Filtering (Mipmapping)

Up level *Down level*

3x3 Square Window (Most Common in image processing)



Horizontal / Vertical Line Window (For Video Coding)

8 Tap

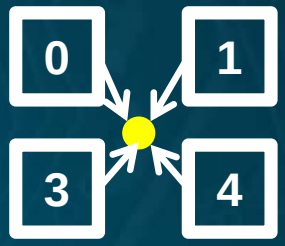
4-tap

Configurability in CFU (II)

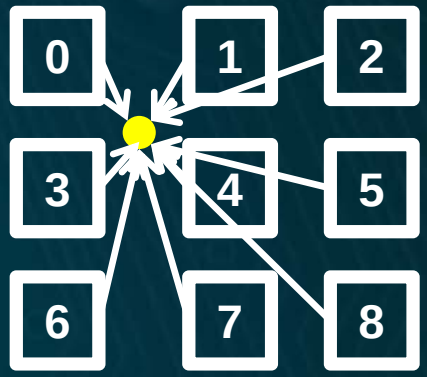
- User-defined FIR filter
  - Flexible sample point with user-defined weighted coefficients.

$$Filter_{FIR}(U,V,D) = \sum_{k \in W(U,V,D)} \{I(k) \cdot C(k)\} + C_{offset}$$

$W(U,V,D)$ = **Point Window**

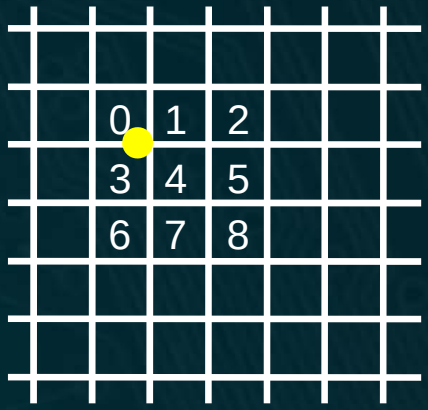


Simple Linear Filter



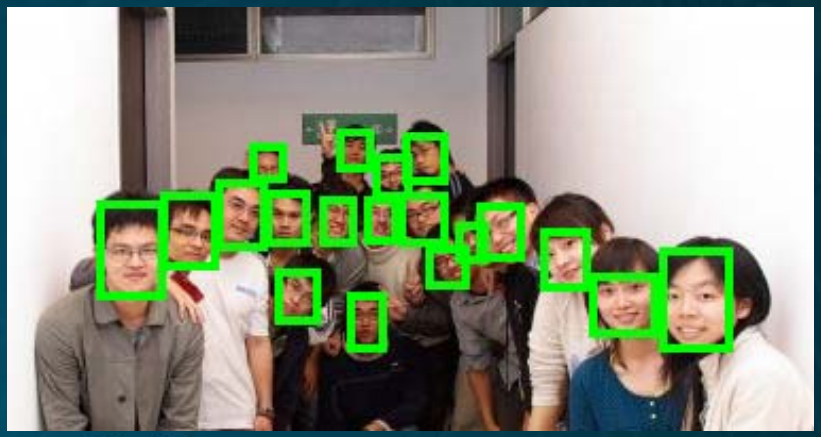
User-defined FIR Filter

Texture Memory



More Applications

- Morphological arithmetics
  - Deal with computer vision applications.
  - Combination of a lot of dilation/erosion operations.
  - Some previous works use GPUs to support real-time processing.
    [R. Yang, Journal of Graphics Tools, Dec 02]



Face Detection

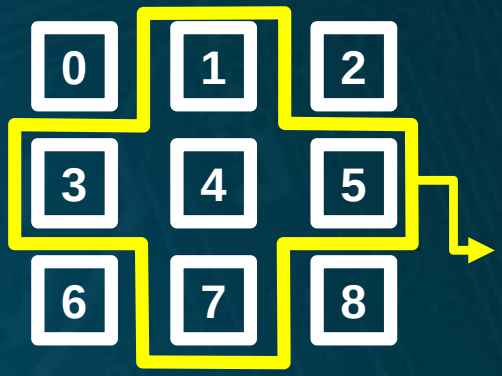
Image Analysis

Configurability in CFU (III)

- User-defined morphological filter
  - Maximum or minimum filter with user-defined structuring element.

$$Filter_{MORPHO}(U,V,D) = Max/min\{ I(k) | C(k) \text{ is enabled}, k \in W(U,V,D)\}$$

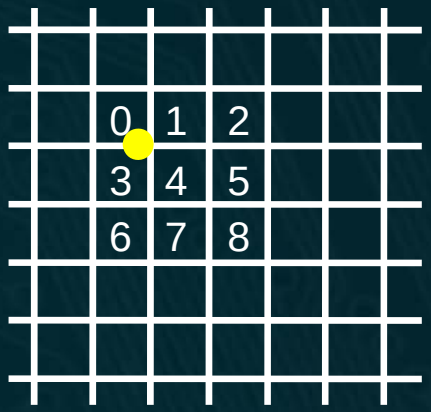
*W(U,V ,D) = **Point Window***



User-defined Irregular Max/min Filter

Enable coefficients, C(k), are used to set different structuring elements in CFU.

Texture Memory



Configurability in CFU (IV)

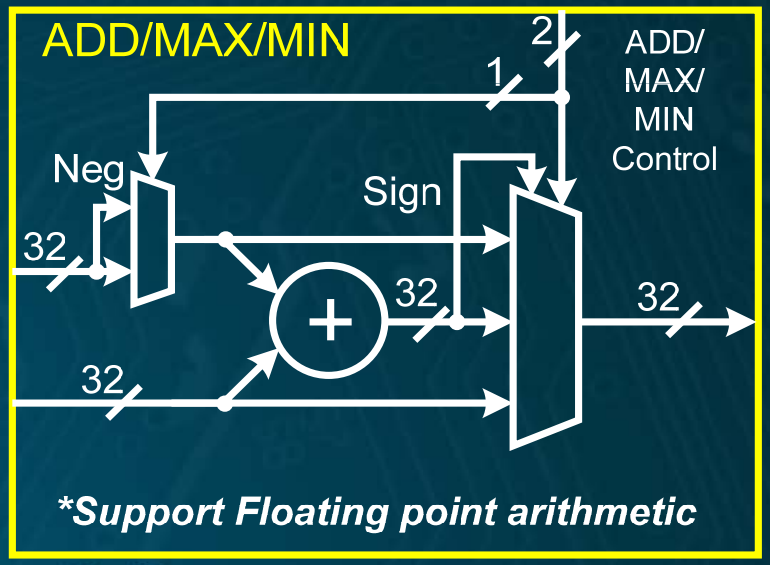
- Multi-sets of coefficients can be selected.
  - Numerous filters with the same sampling shape but different coefficients.
- Take H.264/AVC deblocking filter for an example.
  - According to boundary strength, different filters with different quantize factors are used to eliminate block effect.

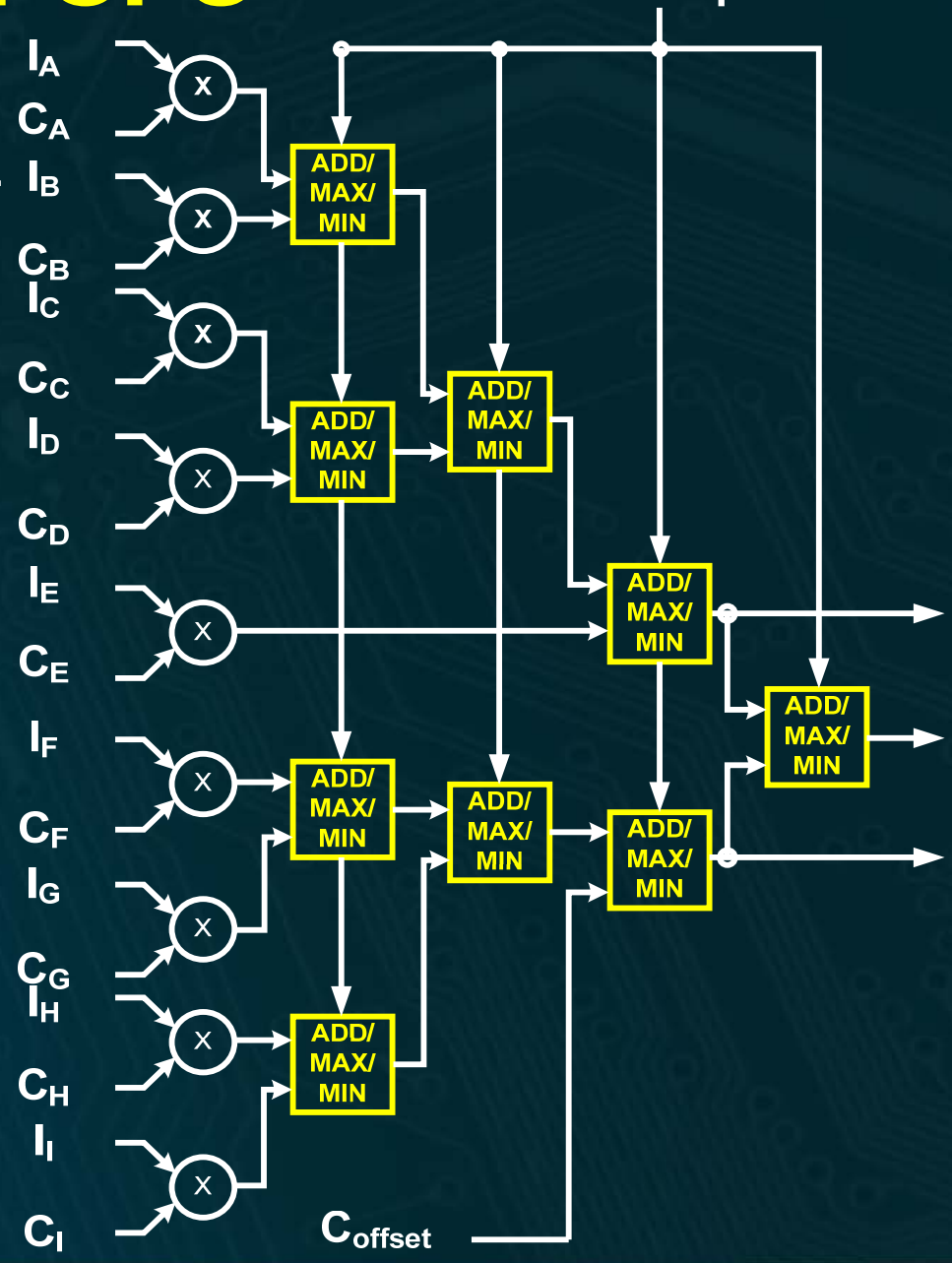
Before deblocking filter

After deblocking filter

Configurable HW of CFU

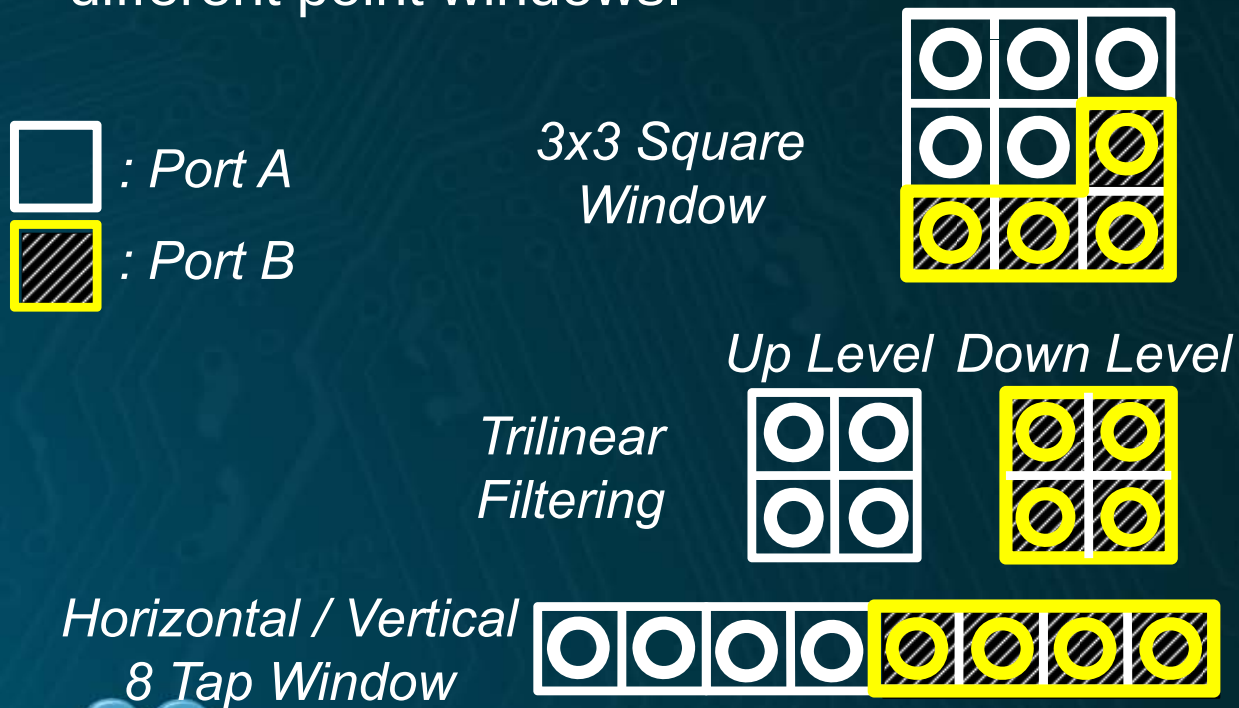
- Execute a nine-tap filtered or two bi-linear filtered samples per cycle.
  - Hardware sharing for low complexity on mobile GPUs.





Caching System

- Anti-conflict storage scheme
  - Adopt two-port on-chip SRAM for simultaneously processing whole data in different point windows.



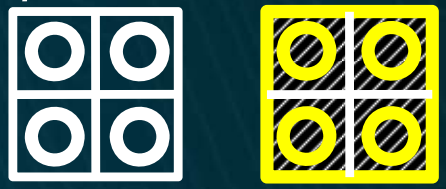
Caching System

- Anti-conflict storage scheme
  - Adopt two-port on-chip SRAM for simultaneously processing whole data in different point windows.

3x3 Square Window

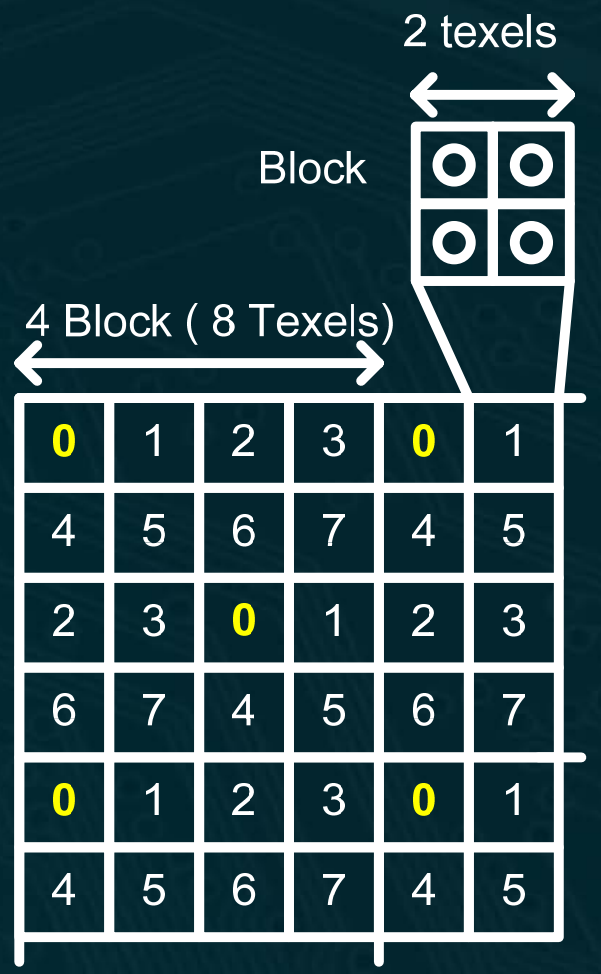
Up Level *Down Level*

Trilinear Filtering



Horizontal / Vertical 8 Tap Window

*Cache line's size is 128 bits. (for **2x2 texels block**)*



Block numbers represent cache bank in SRAM

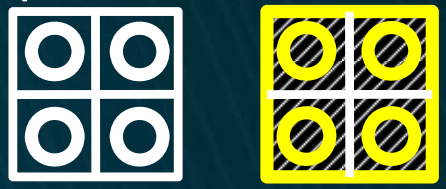
Caching System

- Anti-conflict storage scheme
  - Adopt two-port on-chip SRAM for simultaneously processing whole data in different point windows.

3x3 Square Window

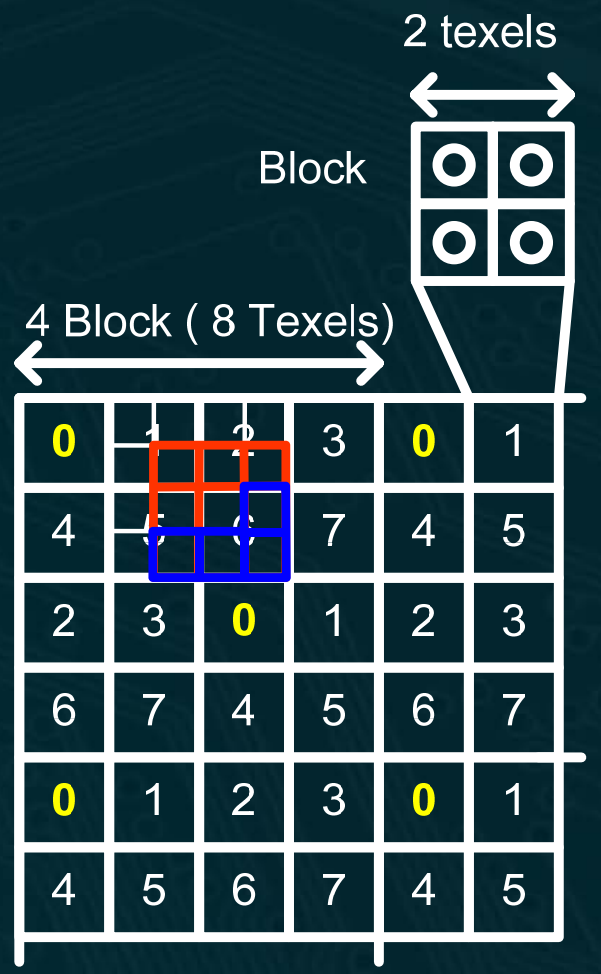
Up Level *Down Level*

Trilinear Filtering



Horizontal / Vertical 8 Tap Window

*Cache line's size is 128 bits. (for **2x2 texels block**)*



Block numbers represent cache bank in SRAM

Caching System

- Anti-conflict storage scheme
  - Adopt two-port on-chip SRAM for simultaneously processing whole data in different point windows.

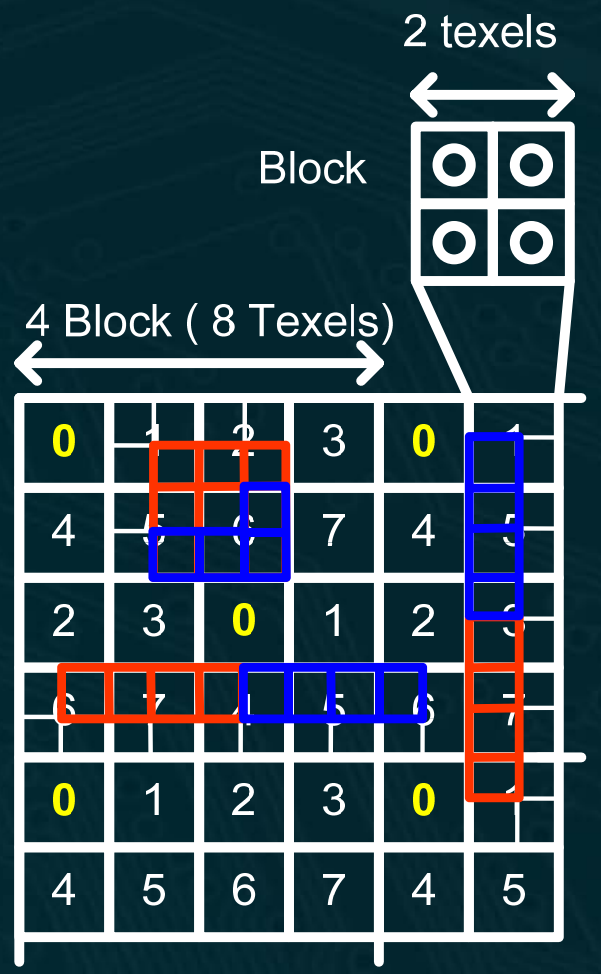
3x3 Square Window

Trilinear Filtering

Up Level *Down Level*

Horizontal / Vertical 8 Tap Window

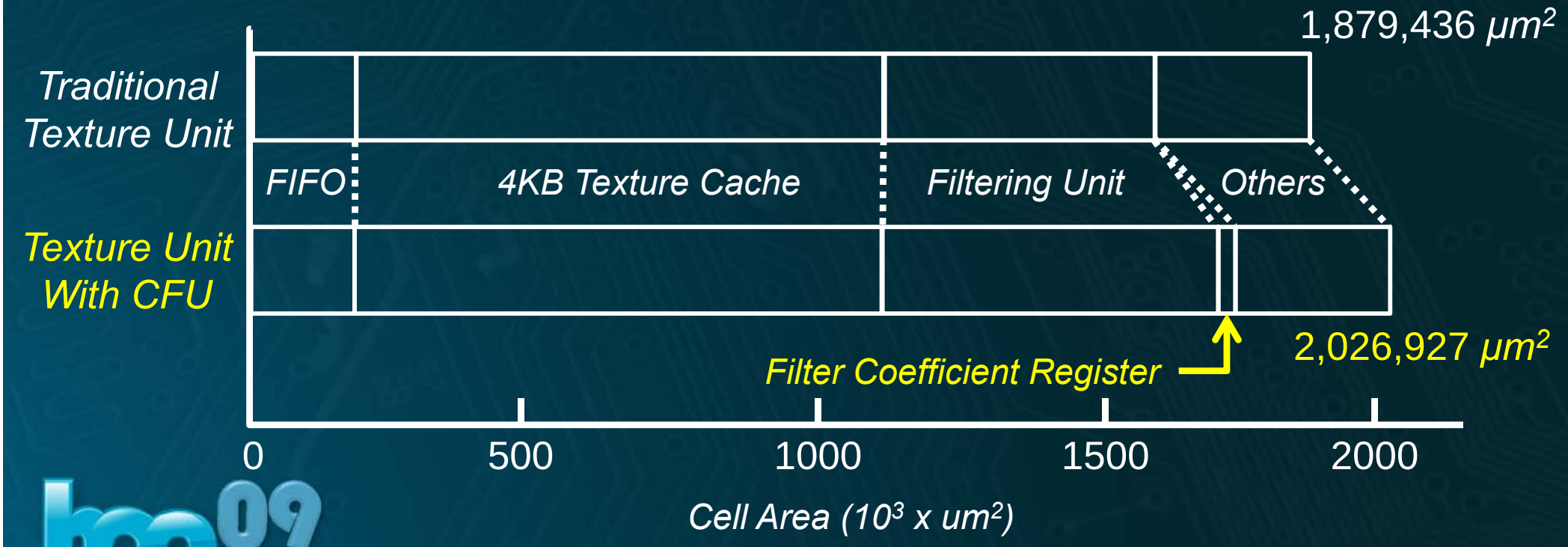
*Cache line's size is 128 bits. (for **2x2 texels block**)*



Block numbers represent cache bank in SRAM

Hardware Overhead

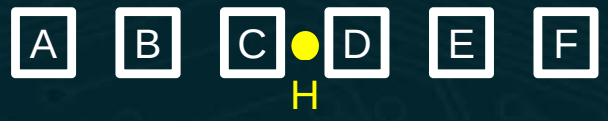
- The working frequency is set to 200MHz on UMC 90nm process with Faraday cell library.
  - The hardware overhead of integrating CFU is only 7.85%.



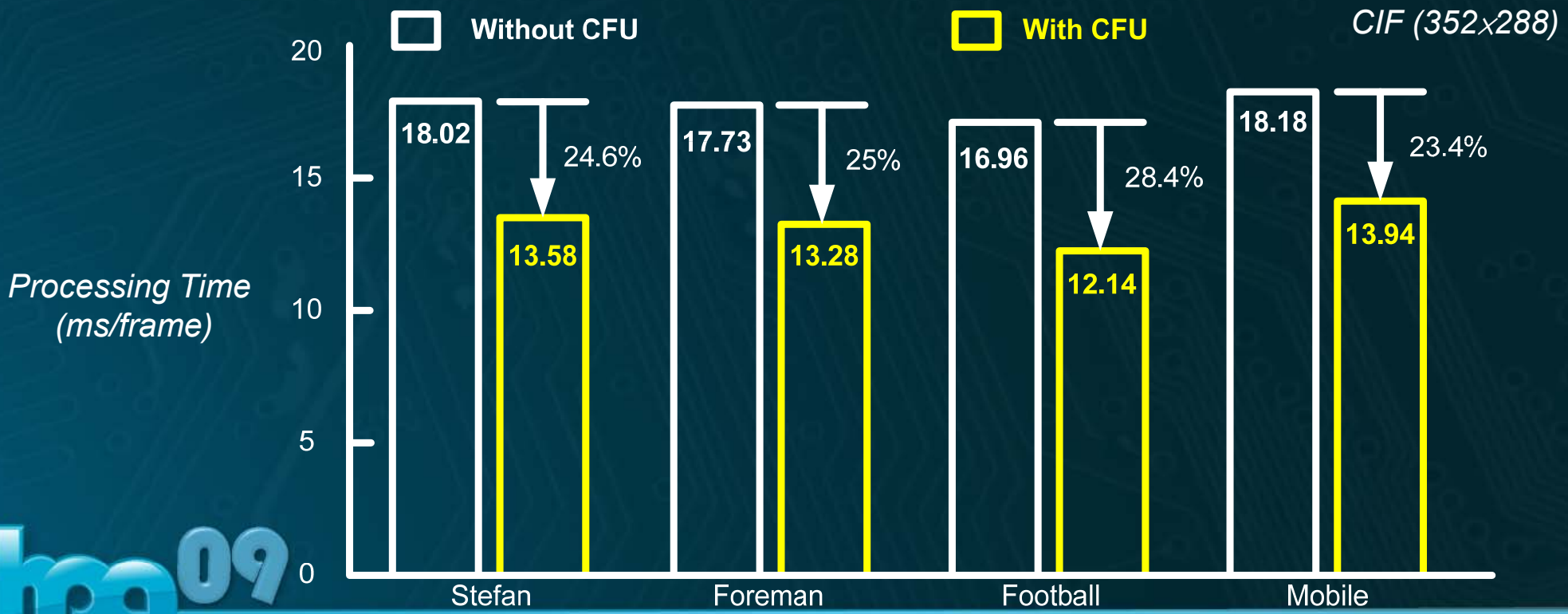
Evaluation Cases (I)

- H.264/AVC motion compensation
  - Apply 6 Tap filter for fractional MC in CFU.

$$H = (A - 5B + 20C + 20D - 5E + F + 16) >> 5$$



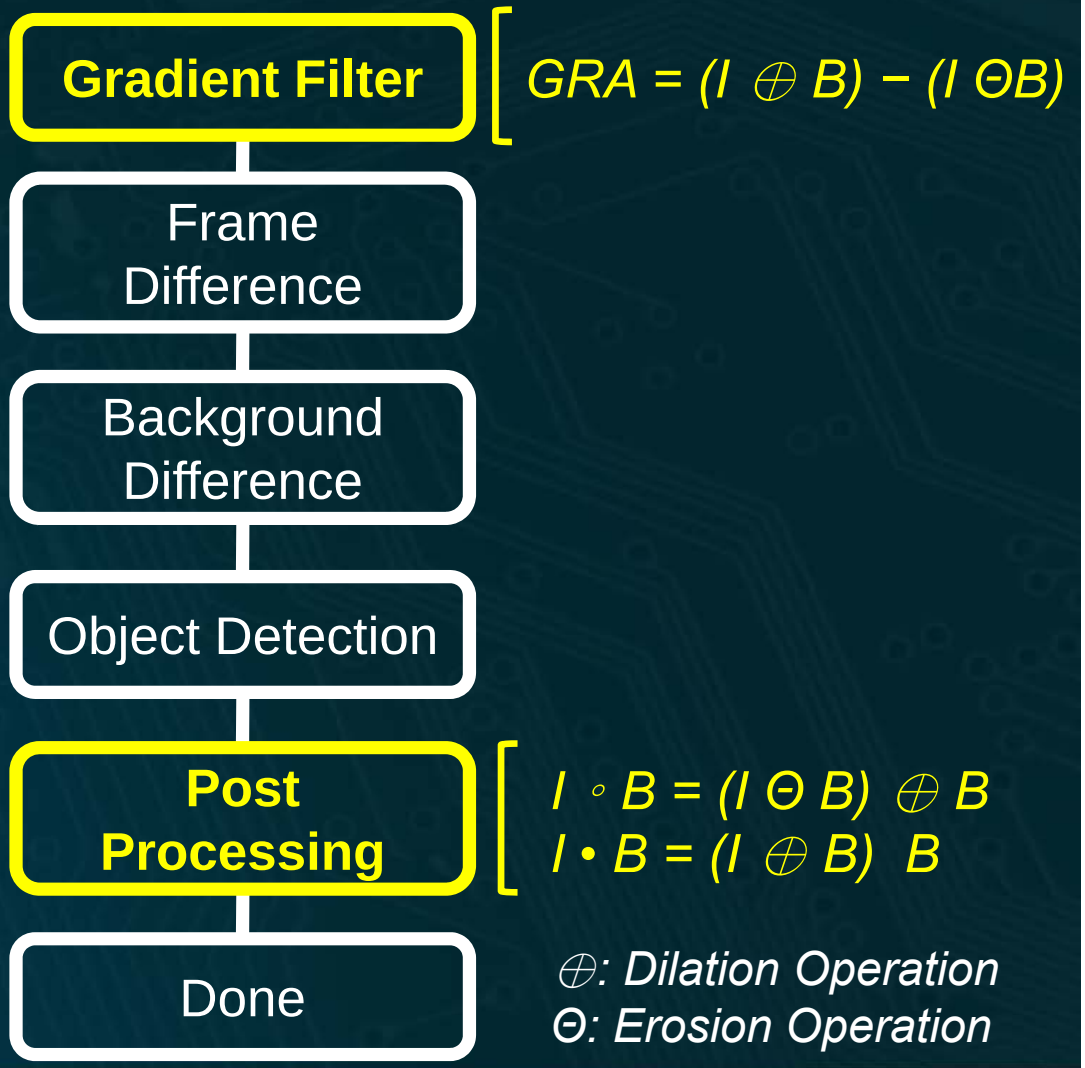
  - 25.35% time is reduced by CFU's efforts.(1.34x faster)



Evaluation Cases (II)

- Chien's fast video segmentation algorithm [S.-Y. Chien, IEEE Transactions on Multimedia, Oct 04]
  - Only apply two subprograms in CFU.
  - The result shows that 58.6% time (73 ms to 30.5 ms) can be reduced on mobile GPUs.

B is the 3 × 3 structuring element of morphological operations.



Conclusion

- CFU provides a new adaptive data accessing method on mobile GPUs, and increases utilization and efficiency of the whole system.
- Simulation results show that processing time can be reduced with CFU.
  - H.264/AVC motion compensation: 25.25% time is saved.
  - Video segmentation algorithm: 58.6% time is saved.
- The hardware overhead of integrating CFU is only 7.8%.

Future Work

- We aim to further increase the capability of the texture unit.
  - Based on the feature of texturing, texture unit supports not only more complexity filtering-like operations, also more efficient flexible accessing methods from external memory by CFU.
  - Become the next generation texture unit model on GPUs.

Acknowledgement

- Thank to Chip Implementation Center (CIC)for EDA tool supports
- Thank to National Science Council (NSC) of Taiwan (R.O.C.) for funding supported.
- Thank to UMC University Program for chip fabrication.

Thank You

Appendix
Control Scheme

- High-level graphics language is required for controlling CFU.

CFU Control Parameters in OpenGL		
Name	Type	Description
TEXTURE MIN FILTER	enum	Set Pixel Window
TEXTURE MAG FILTER	enum	Set Pixel Window
TEXTURE FILTER(**x**) TYPE	enum	Set Filter Type (FIR, MAX, MIN)
TEXTURE FILTER(**x**) COEF	10 floats	Set weighting coef. or enable coef.
CFU Control Function in Cg		
Tex2D CFU(sampler2D tex, float2 st, int **UserFilterID**)		

***X** : Indicates numerous sets of filtering parameters*
***UserFilterID** : Indicates which user-defined filter is called*