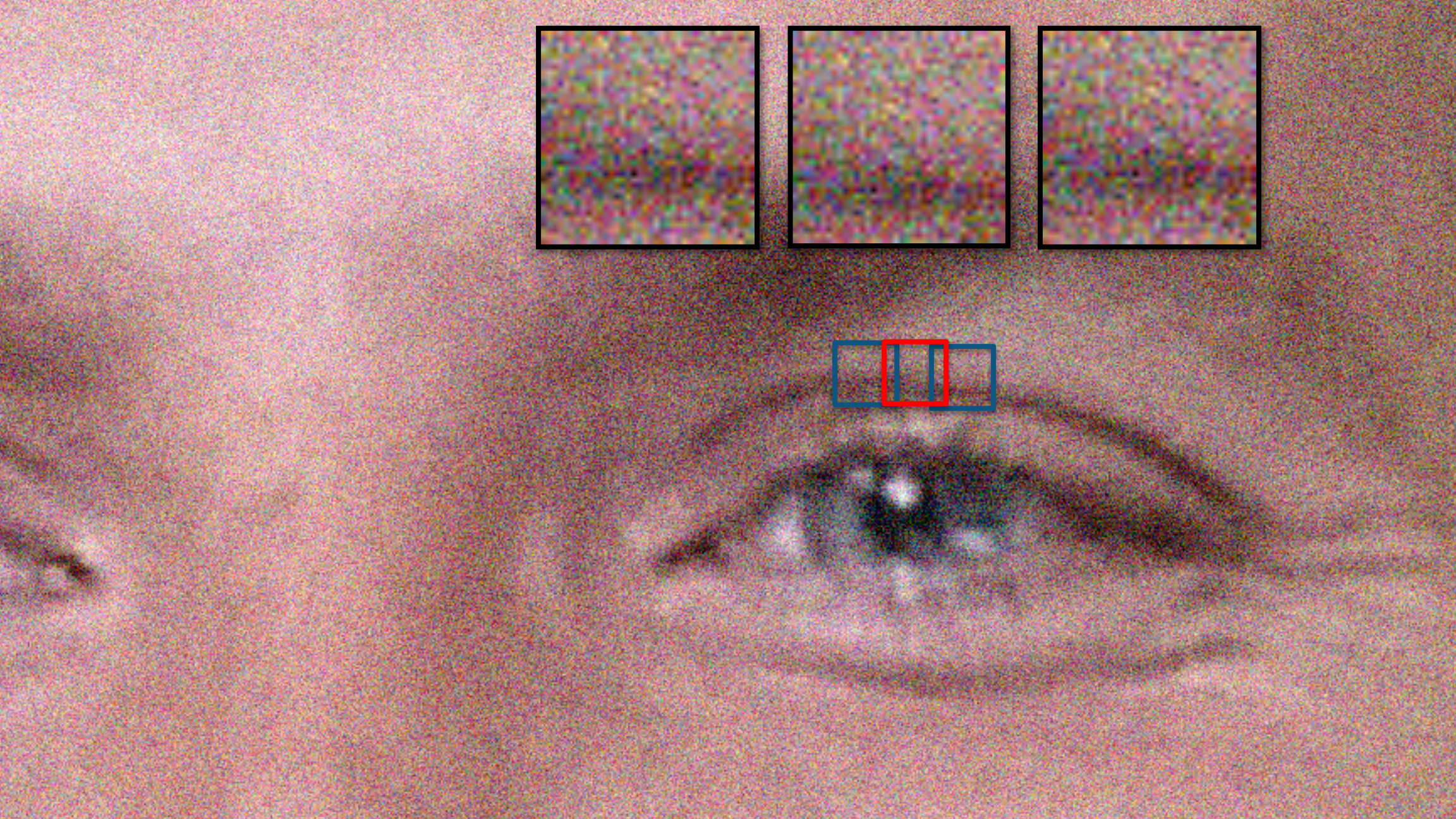
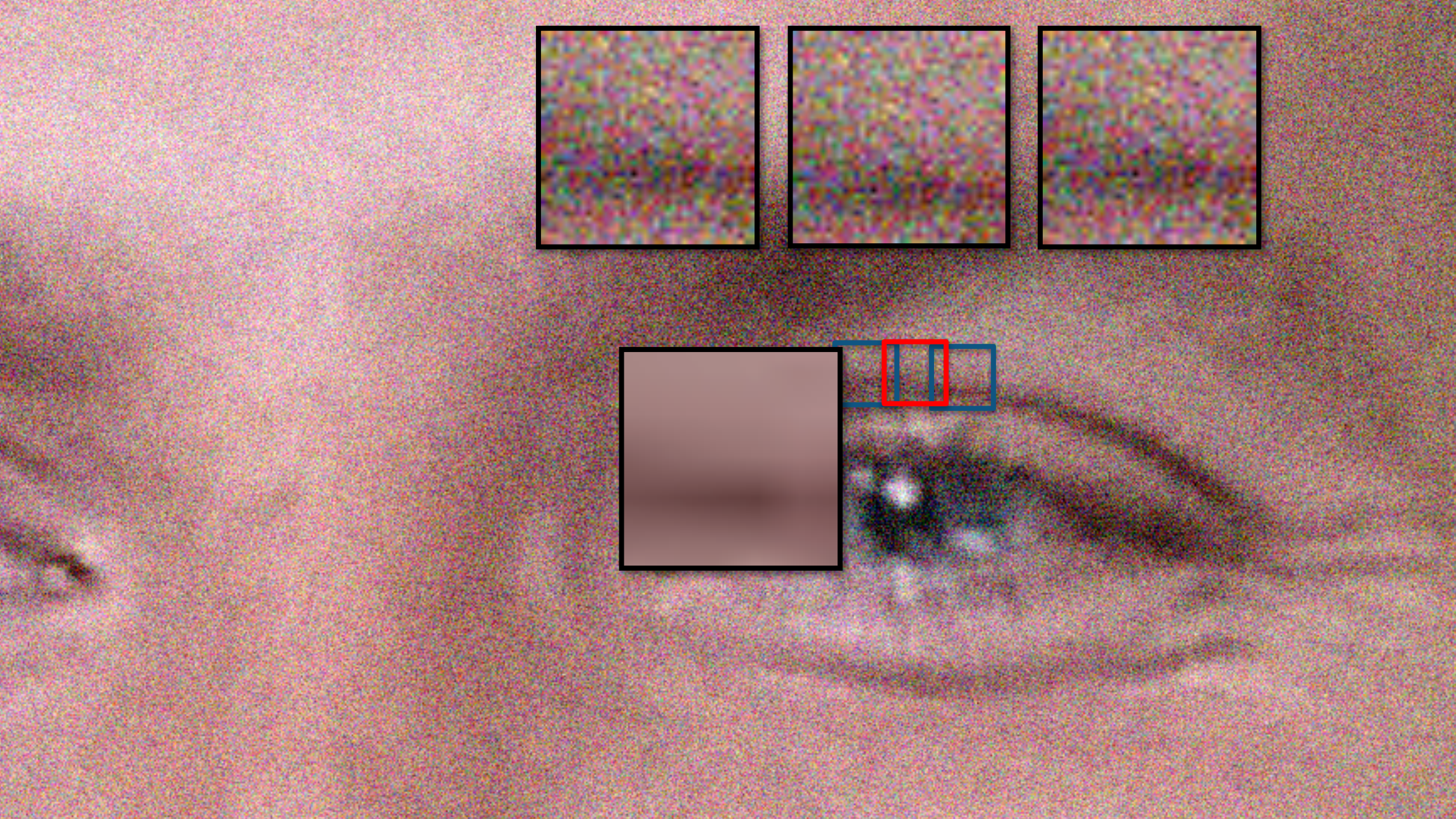
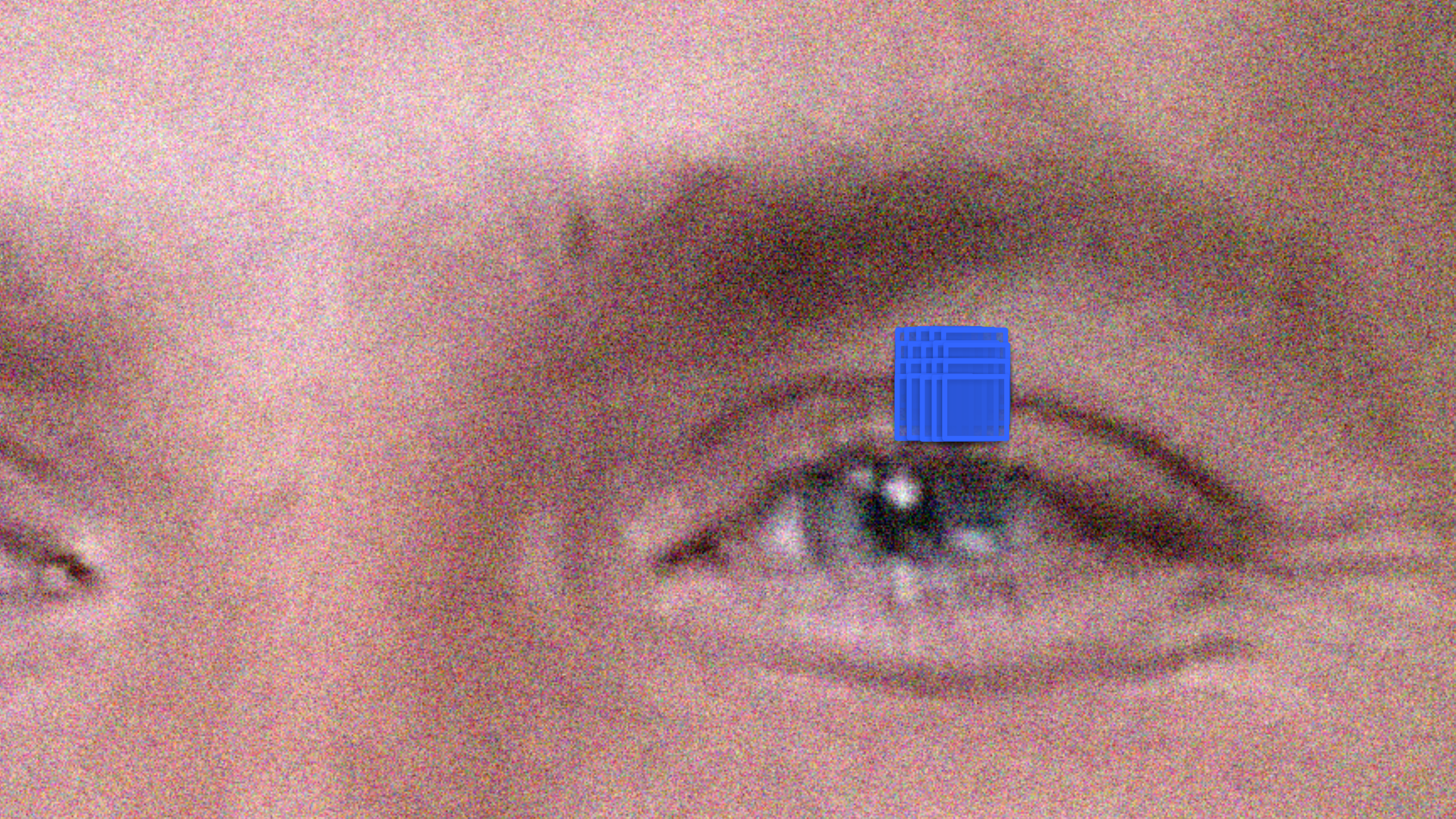
NVIDIA

FASTANN FOR HIGH QUALITY COLLABORATIVE FILTERING

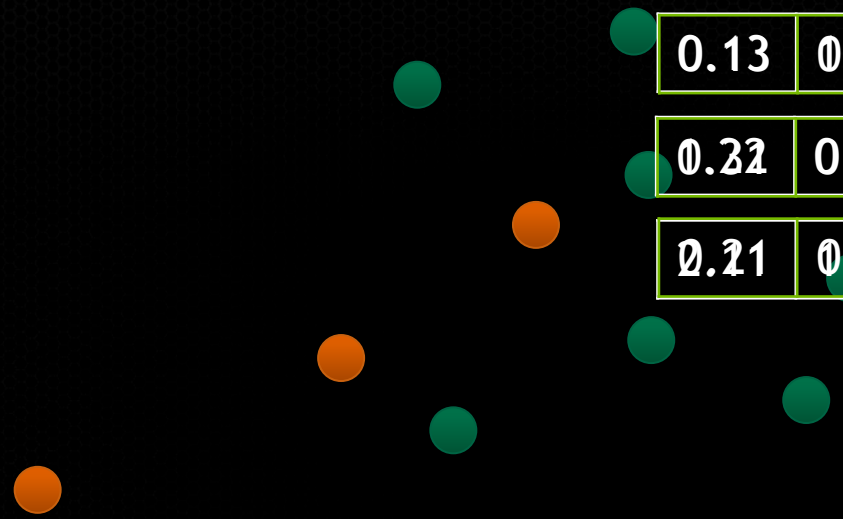
Yun-Ta Tsai, Markus Steinberger, Dawid Pająk, Kari Pulli







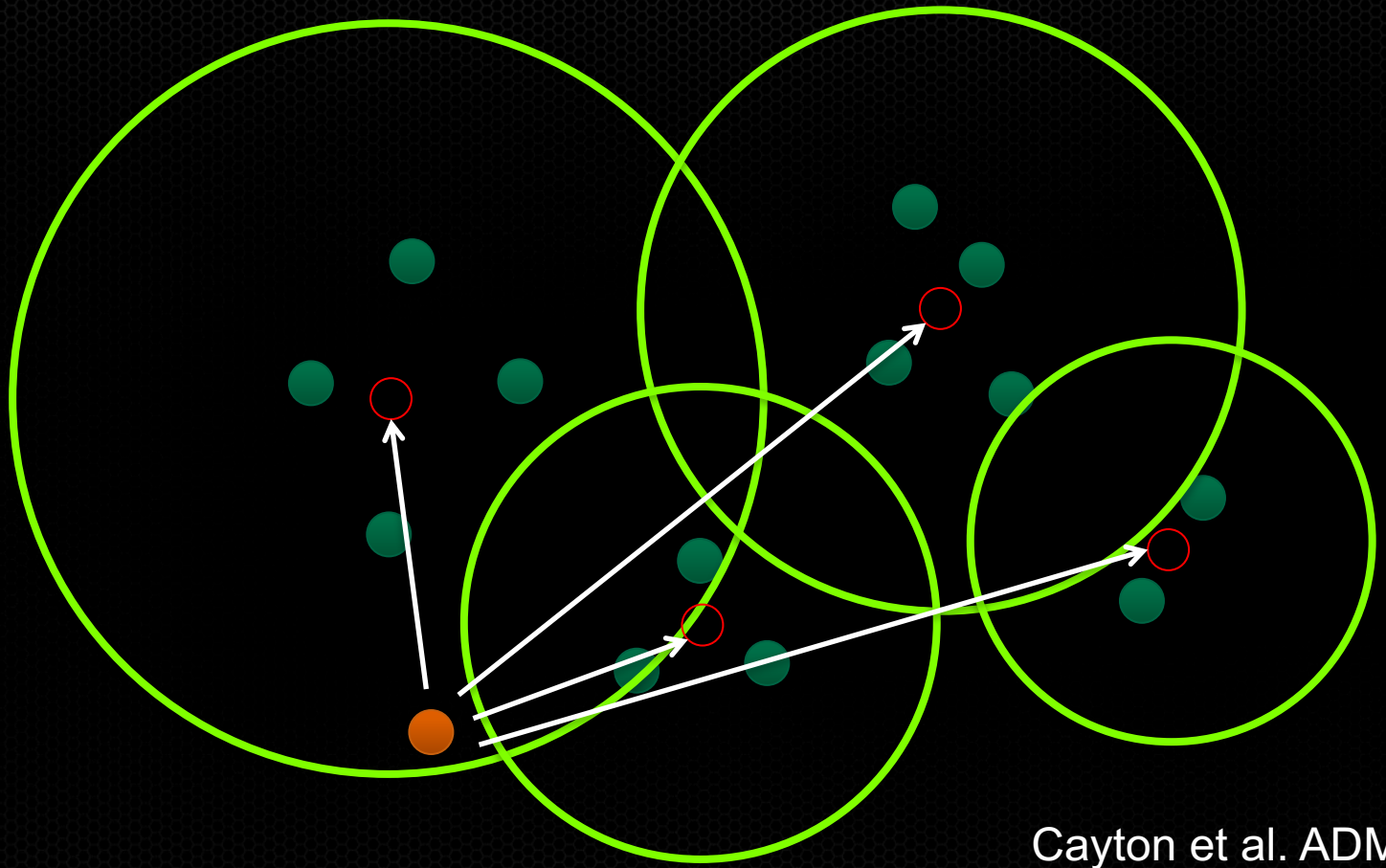
Related Work



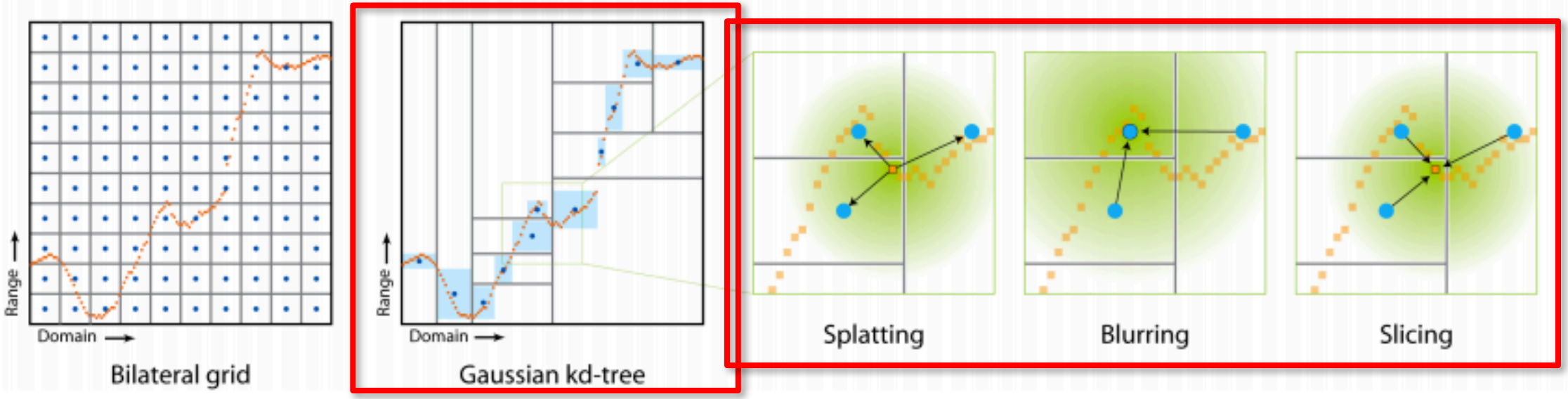
Distance Table (sorted)

0.13	0.48	0.98	1.33	1.41	2.33
0.22	0.45	0.48	2.02	1.75	0.98
0.21	0.32	0.92	3.16	0.33	0.16

Garcia et al. ICIP 2010

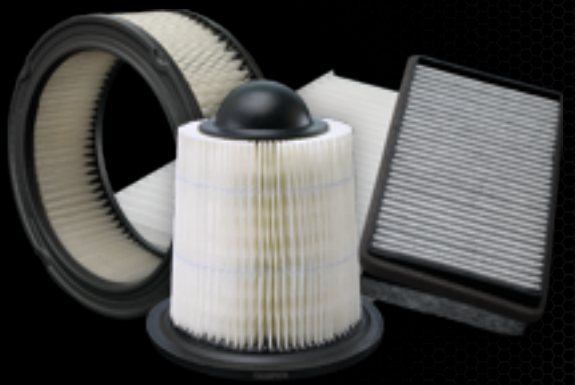


Cayton et al. ADMS 2010

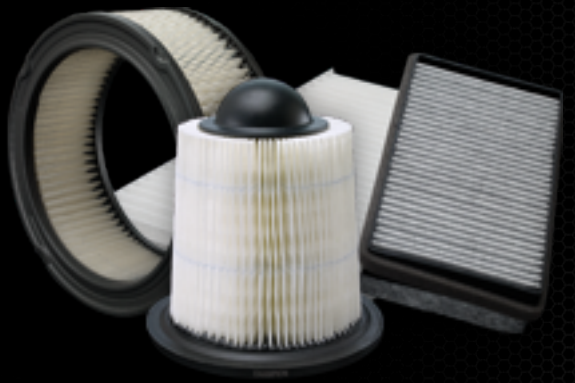


Adams et al. SIGGRAPH 2009

Limitation

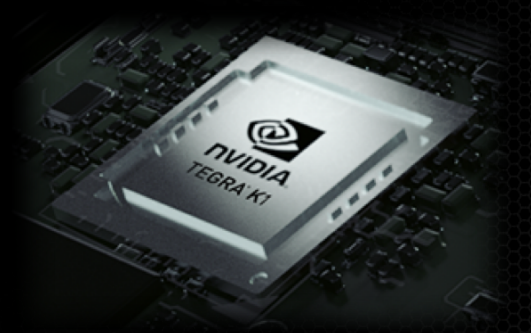


Our solution



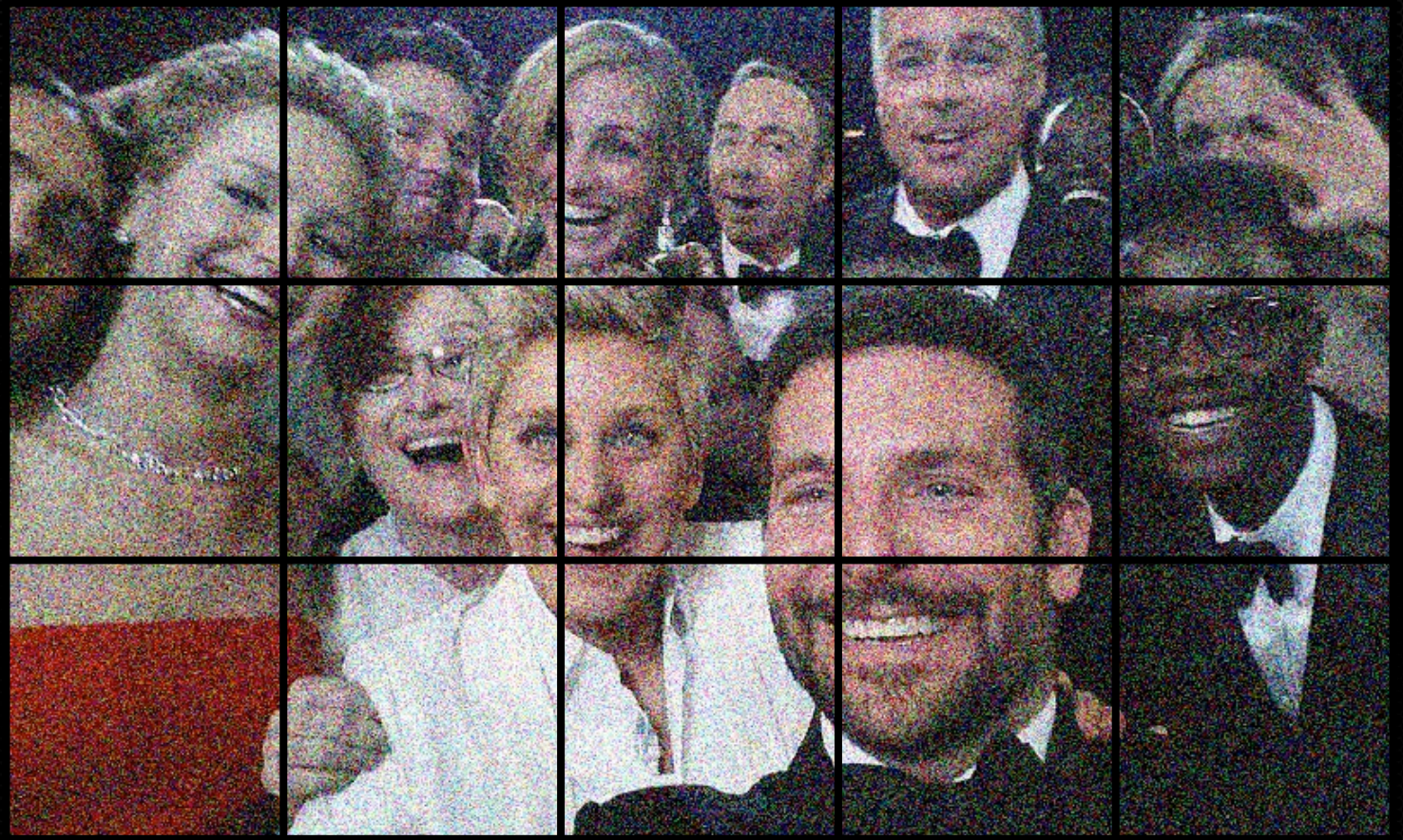
Our solution

- Efficient implementation on GPU
- General solution for different filters
- High image quality
- Applicable to different applications



NVIDIA

Our solution



Our solution

Clustering

kNN Query

Filtering

Design challenges

- Register pressure
- Memory access pattern
- Thread divergence
- Kernel launch overhead
- Memory footprint

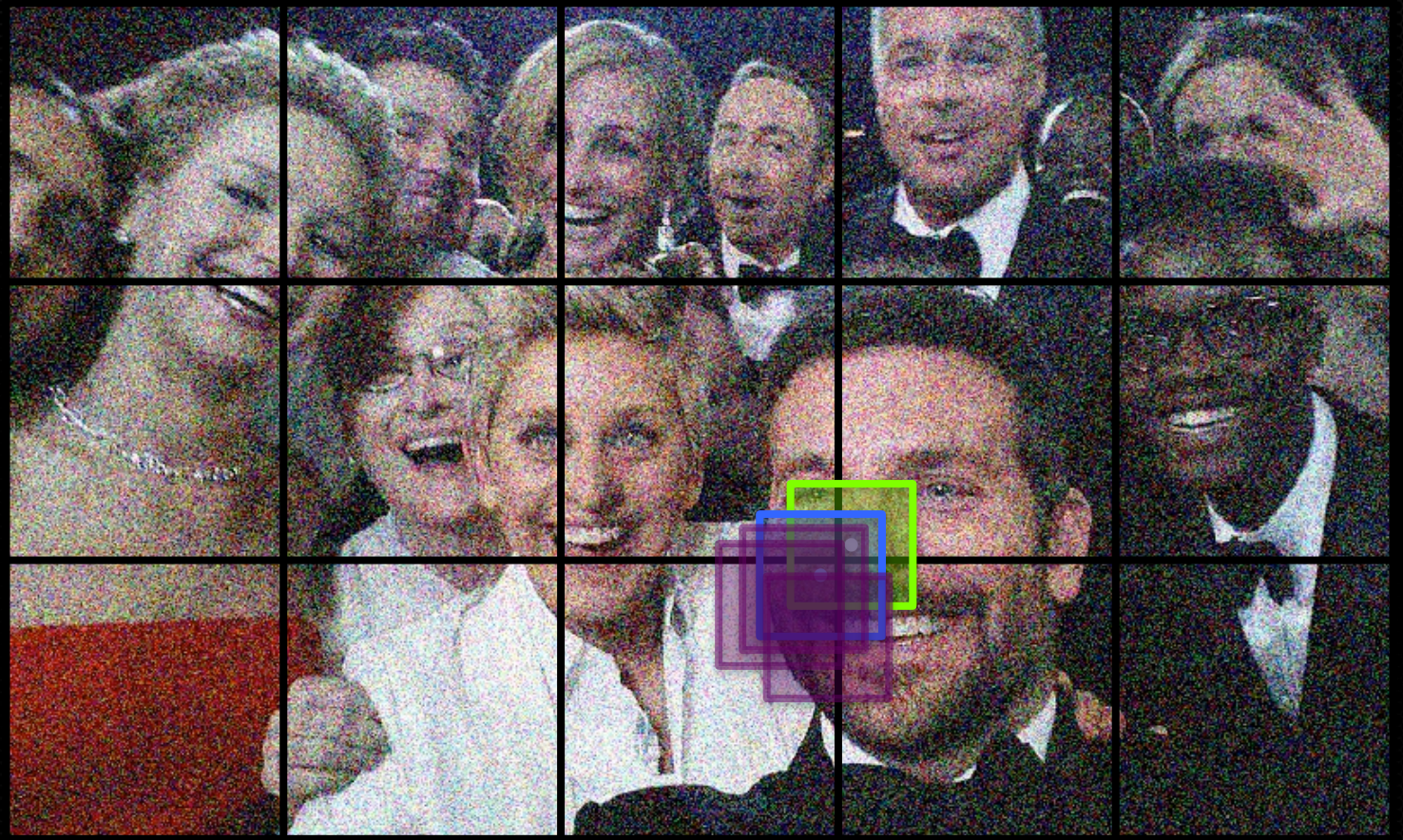
Our solution

Clustering

kNN Query

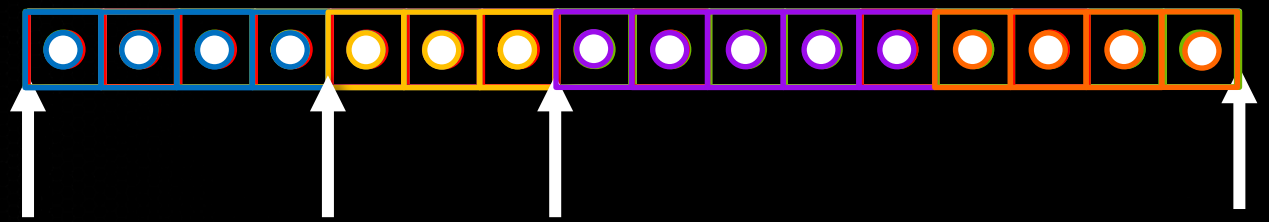
Filtering

Tiling

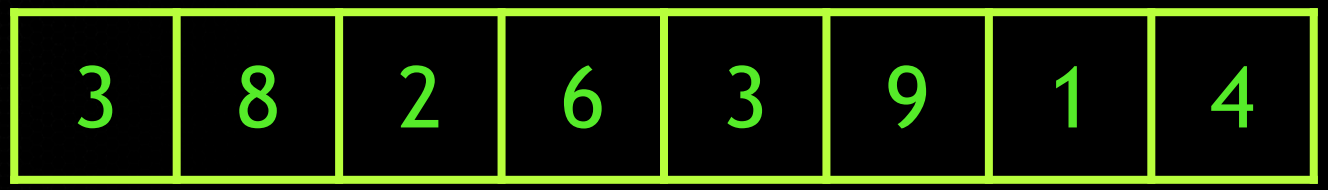


Clustering

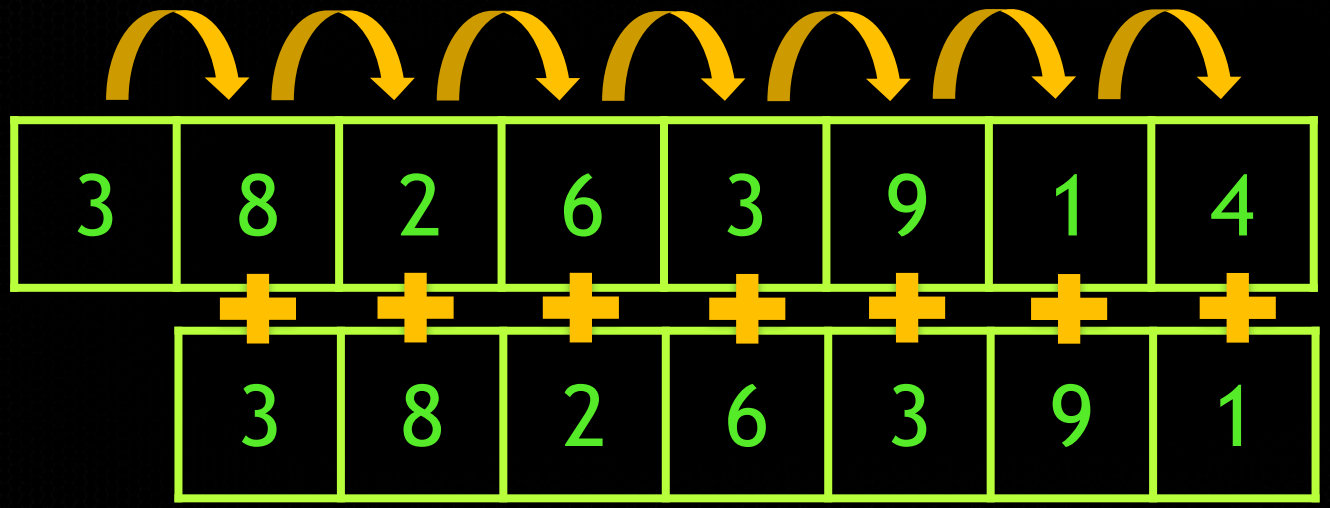
Clustering



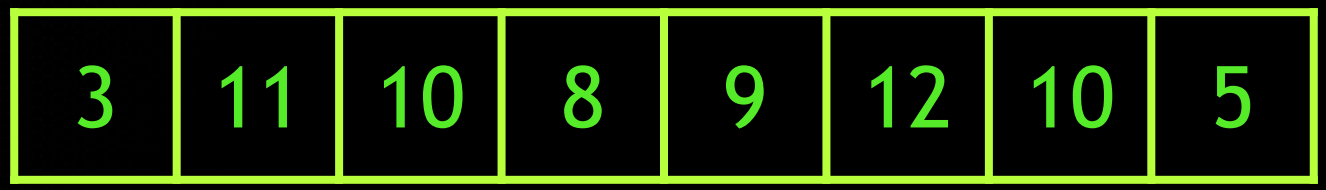
Warp-wide operation



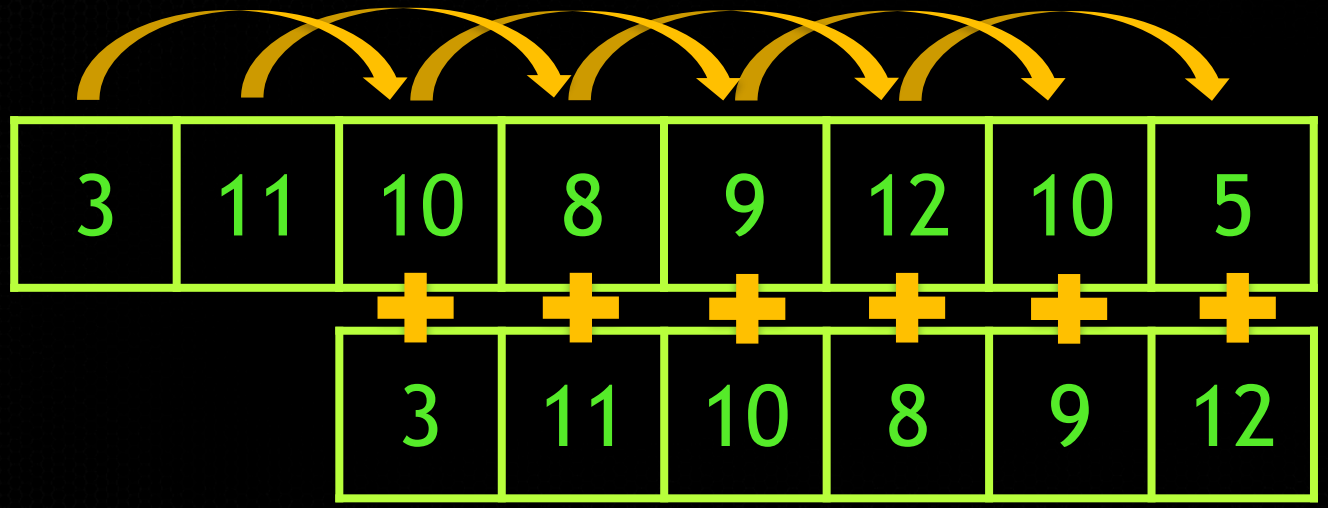
Warp-wide operation



Warp-wide operation

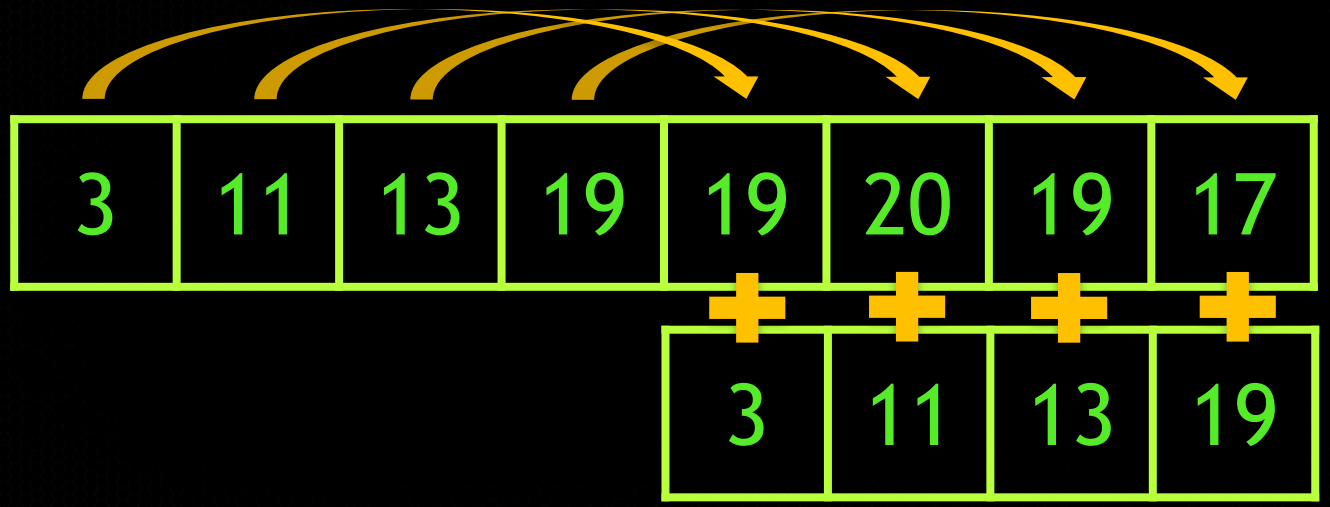


Warp-wide operation

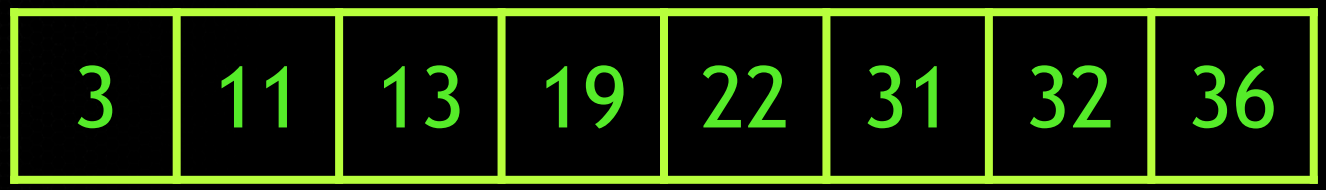


Warp-wide operation

Warp-wide operation

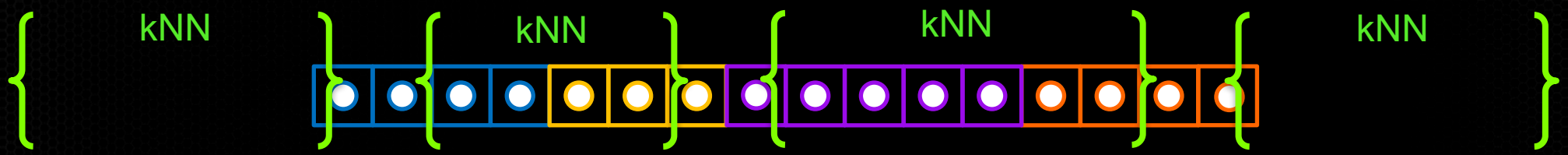


Warp-wide operation



- Reduce register usage
- Better parallelism
- Minimize thread and warp divergence

Clustering



Our solution

Clustering

kNN Query

Filtering

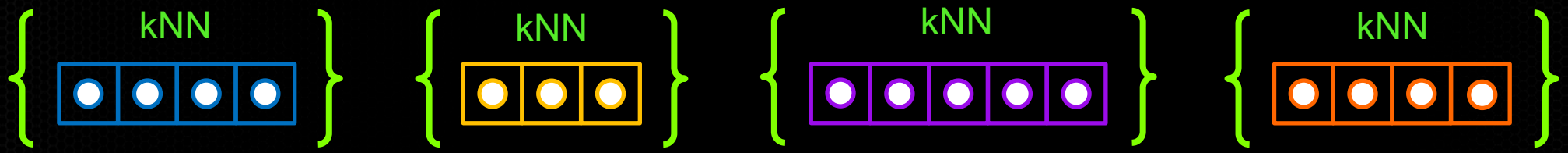
Our solution

Clustering

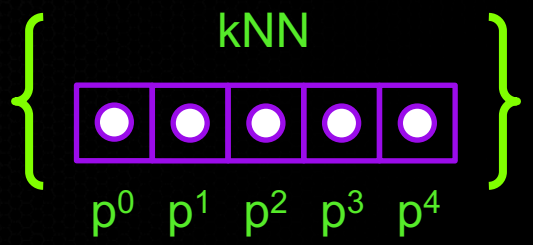
kNN Query

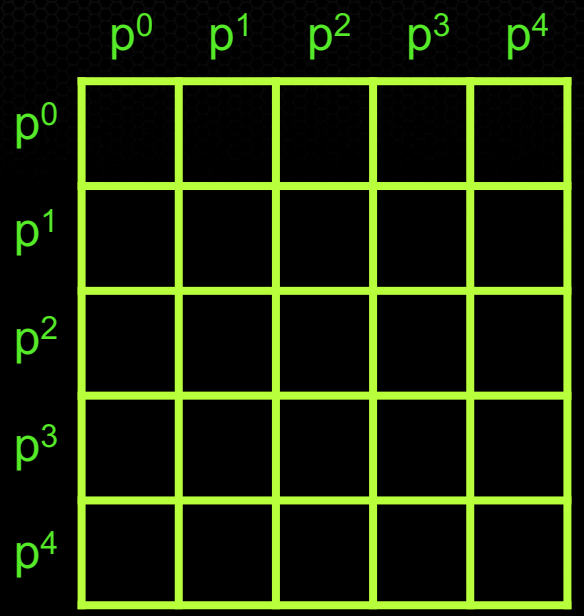
Filtering

kNN Query

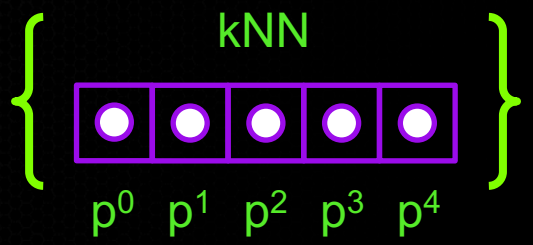


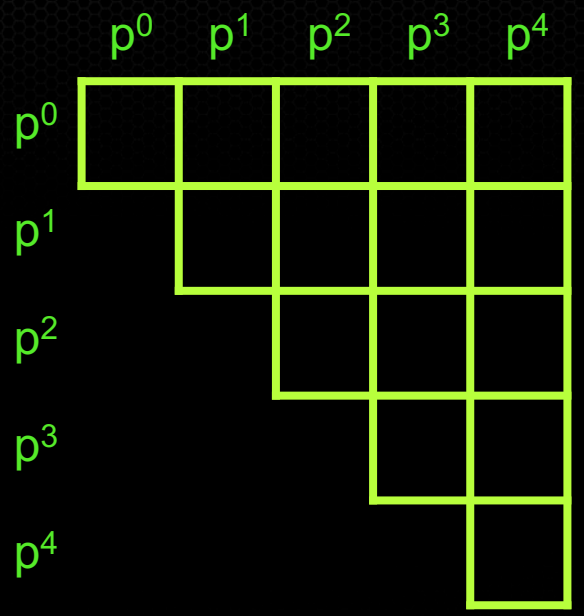
kNN Query



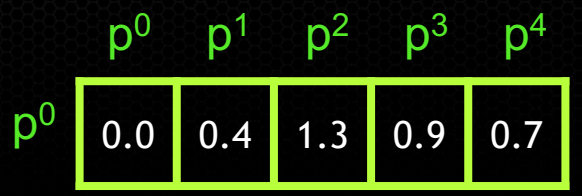


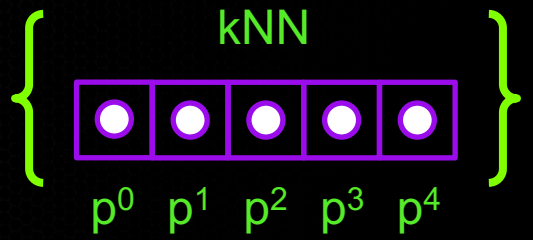
kNN Query





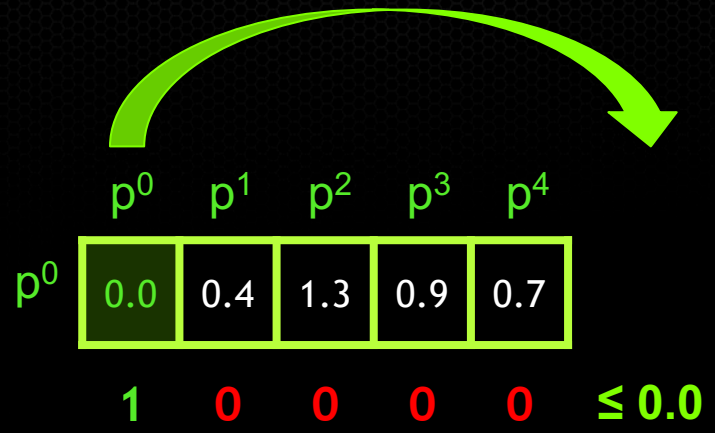
kNN Query



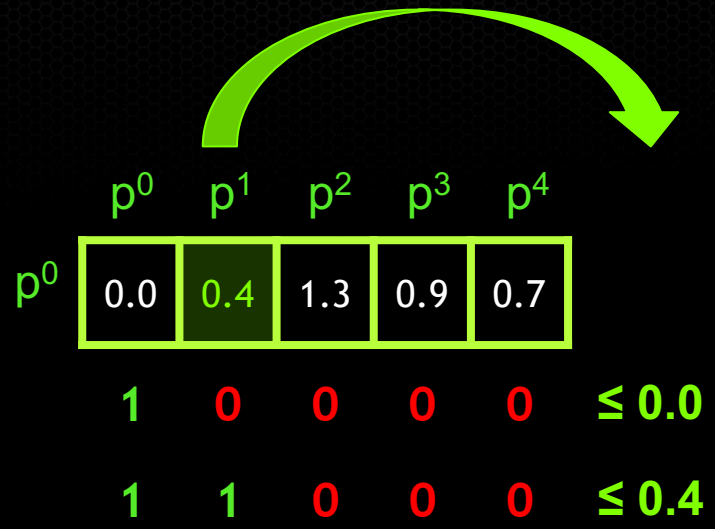


kNN Query

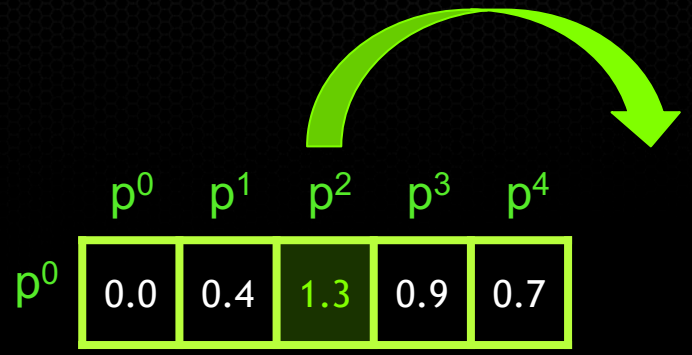
kNN Query



kNN Query

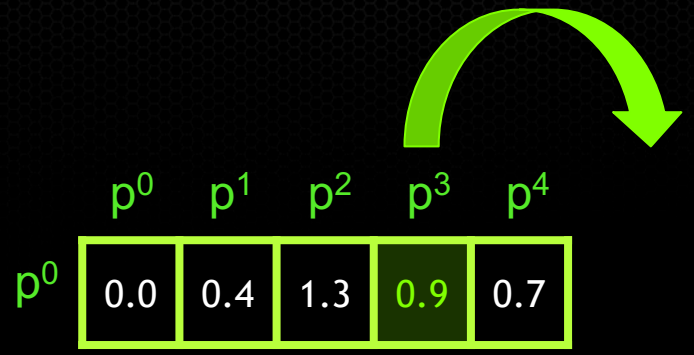


kNN Query



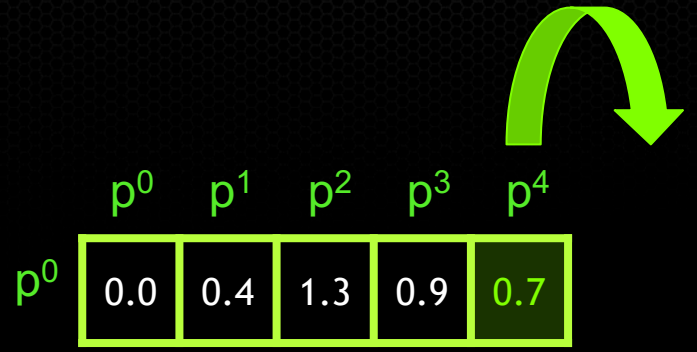
1	0	0	0	0	≤ 0.0
1	1	0	0	0	≤ 0.4
1	1	1	1	1	≤ 1.3

kNN Query



1	0	0	0	0	≤ 0.0
1	1	0	0	0	≤ 0.4
1	1	1	1	1	≤ 1.3
1	1	0	1	1	≤ 0.9

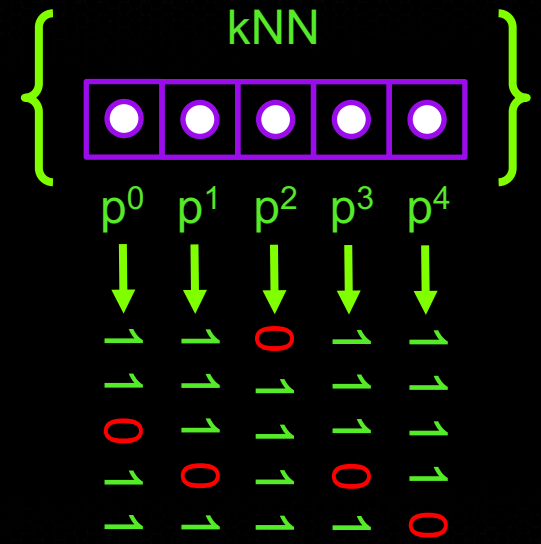
kNN Query



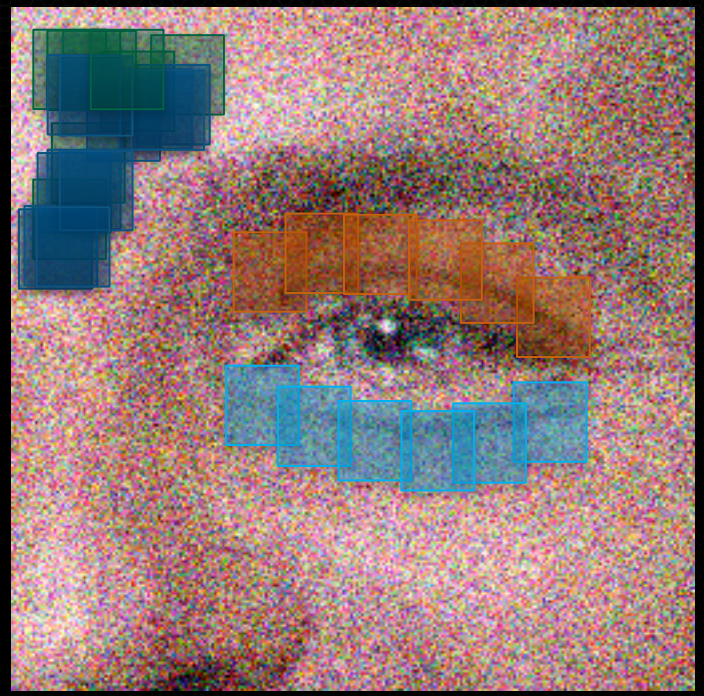
1	0	0	0	0	≤ 0.0
1	1	0	0	0	≤ 0.4
1	1	1	1	1	≤ 1.3
1	1	0	1	1	≤ 0.9
1	1	0	0	1	≤ 0.7

kNN Query

kNN Query

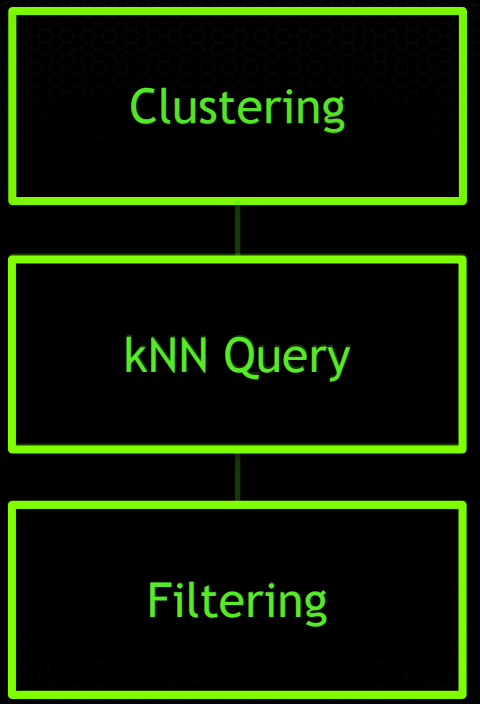


Filtering



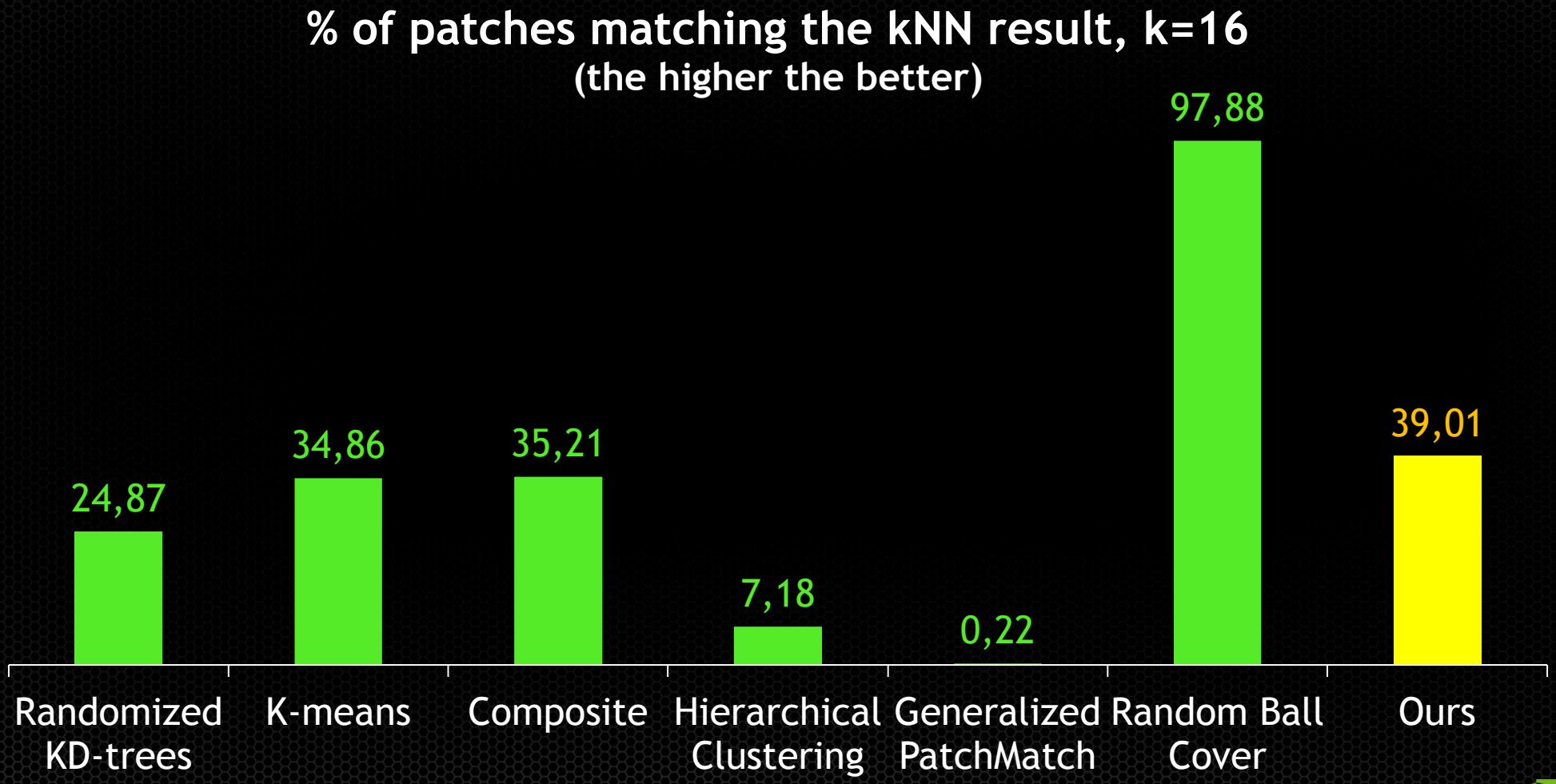
Our solution

- Hierarchical 2-mean clustering
- Warp-wide operators
- kNN search
- Distance table
- Voting
- Binary coding
- Filtering and aggregation



Results

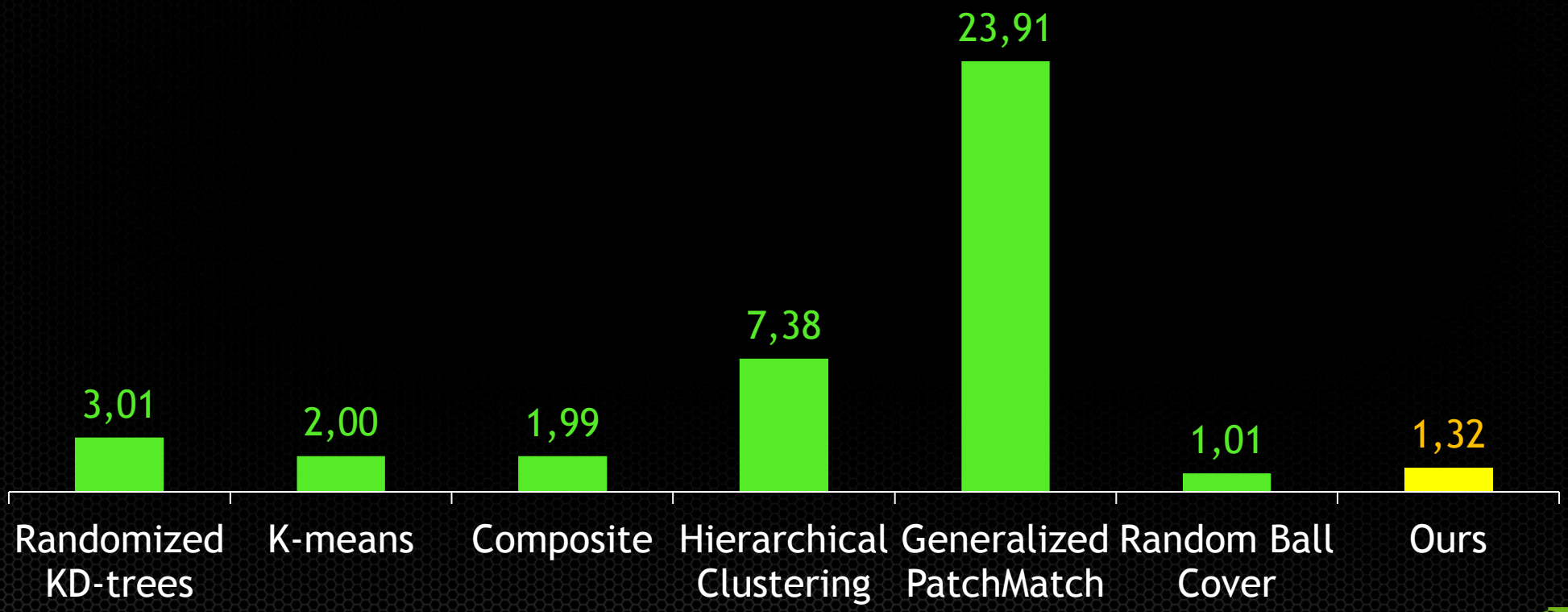
NN quality



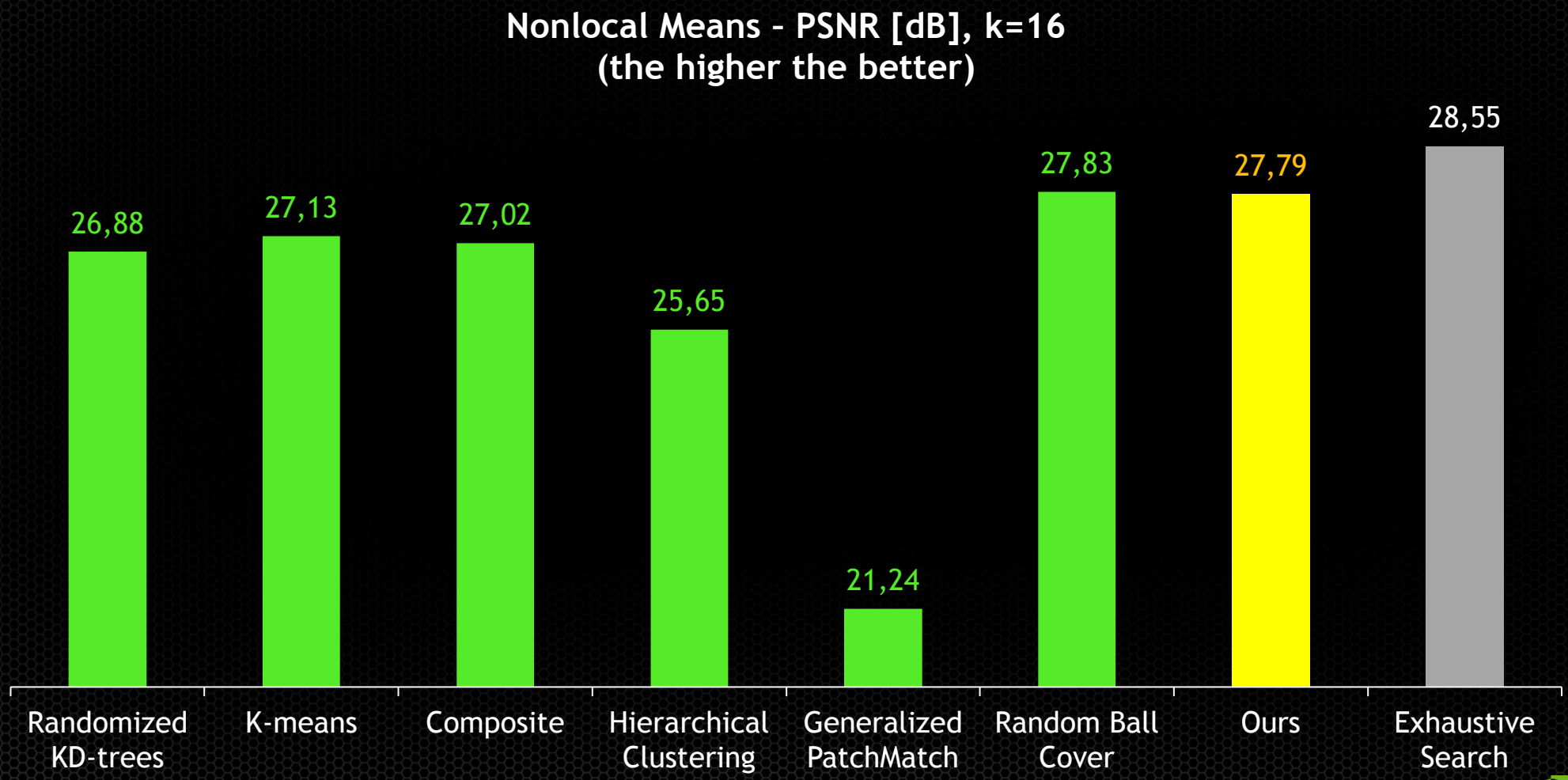
NVIDIA

NN quality

D_{ann}/D_{knn}, k=16
(the lower the better)

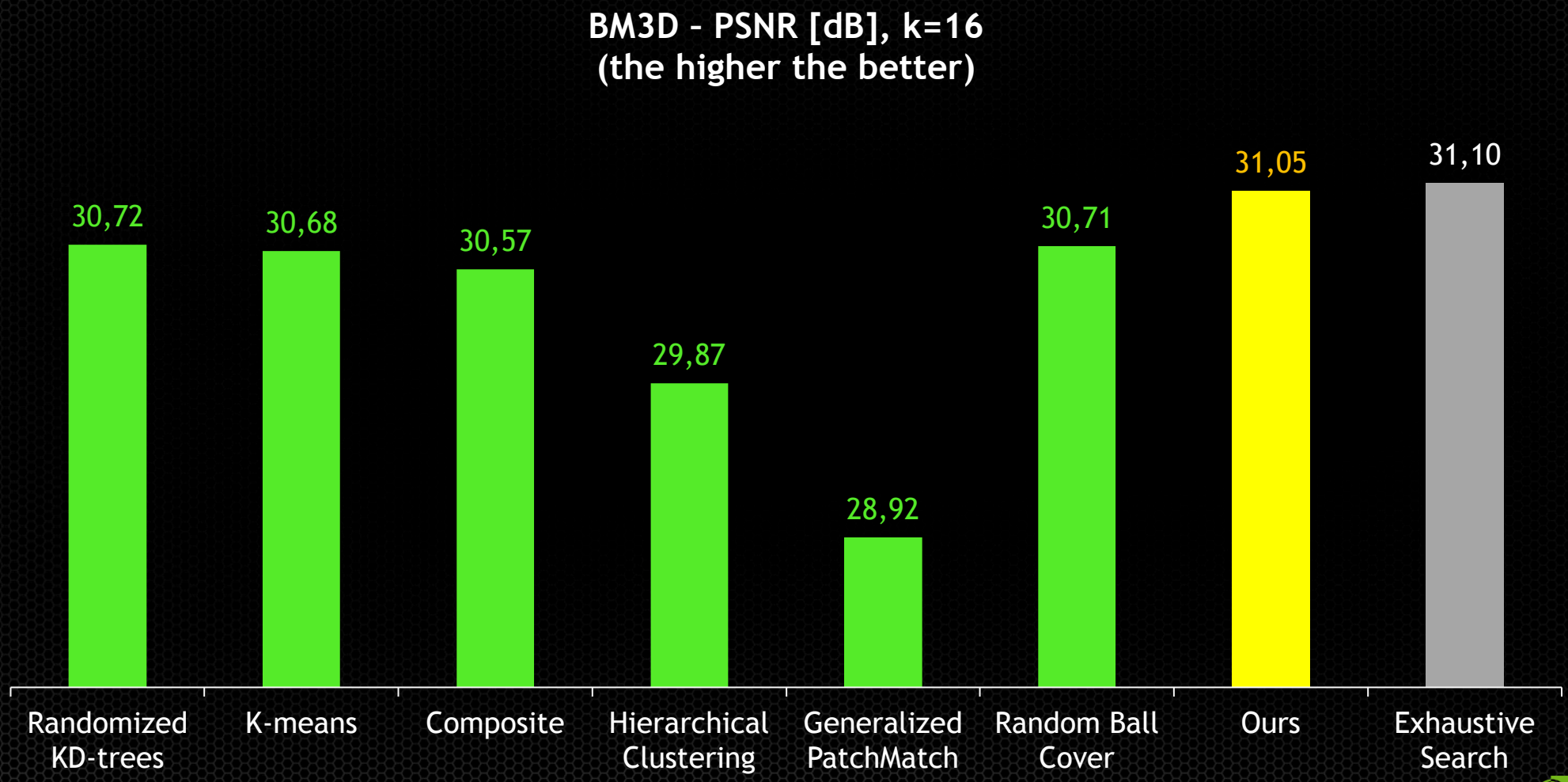


Single Frame Noise Reduction



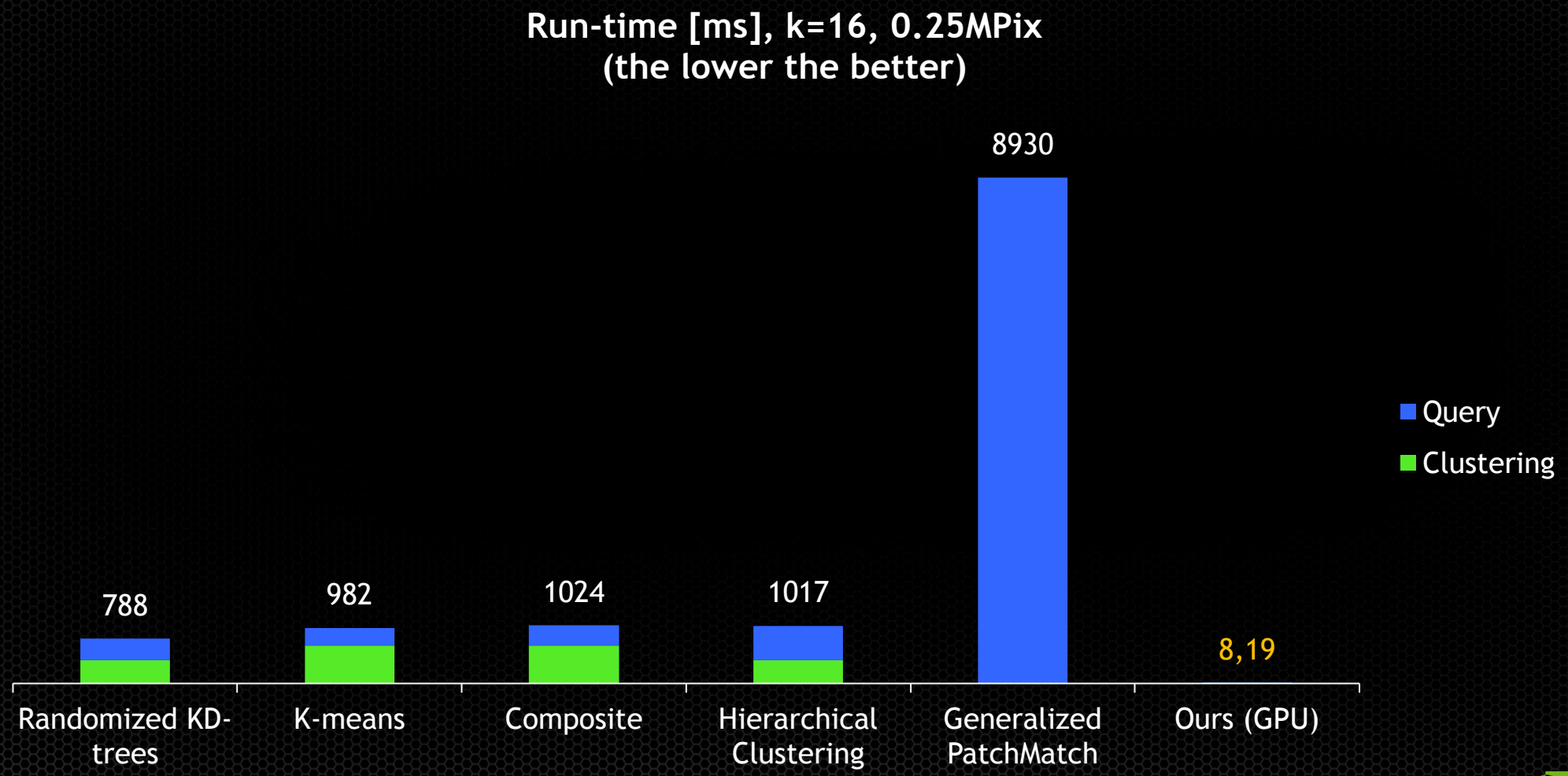
NVIDIA

Single Frame Noise Reduction

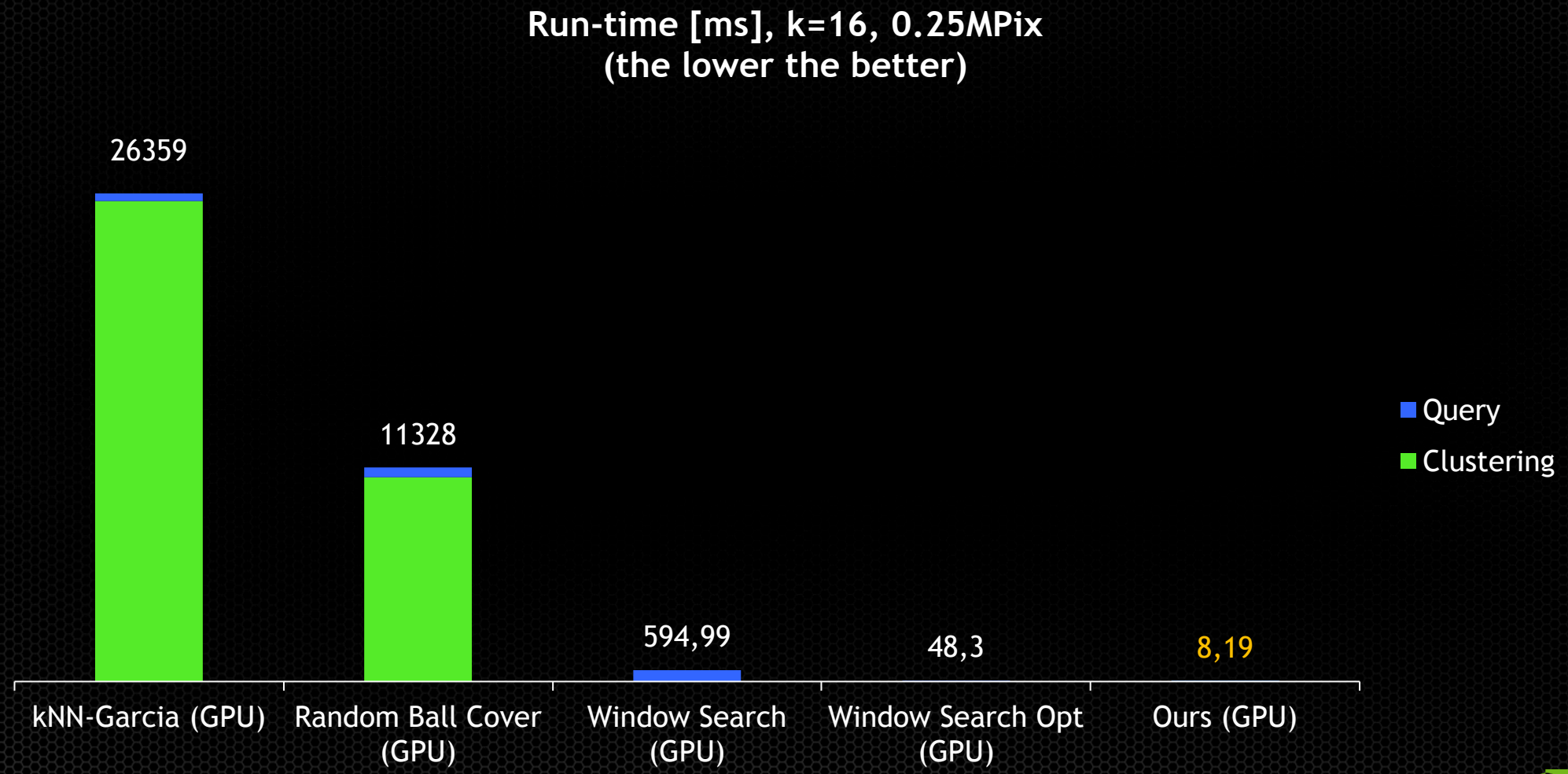


NVIDIA

Single Frame Noise Reduction



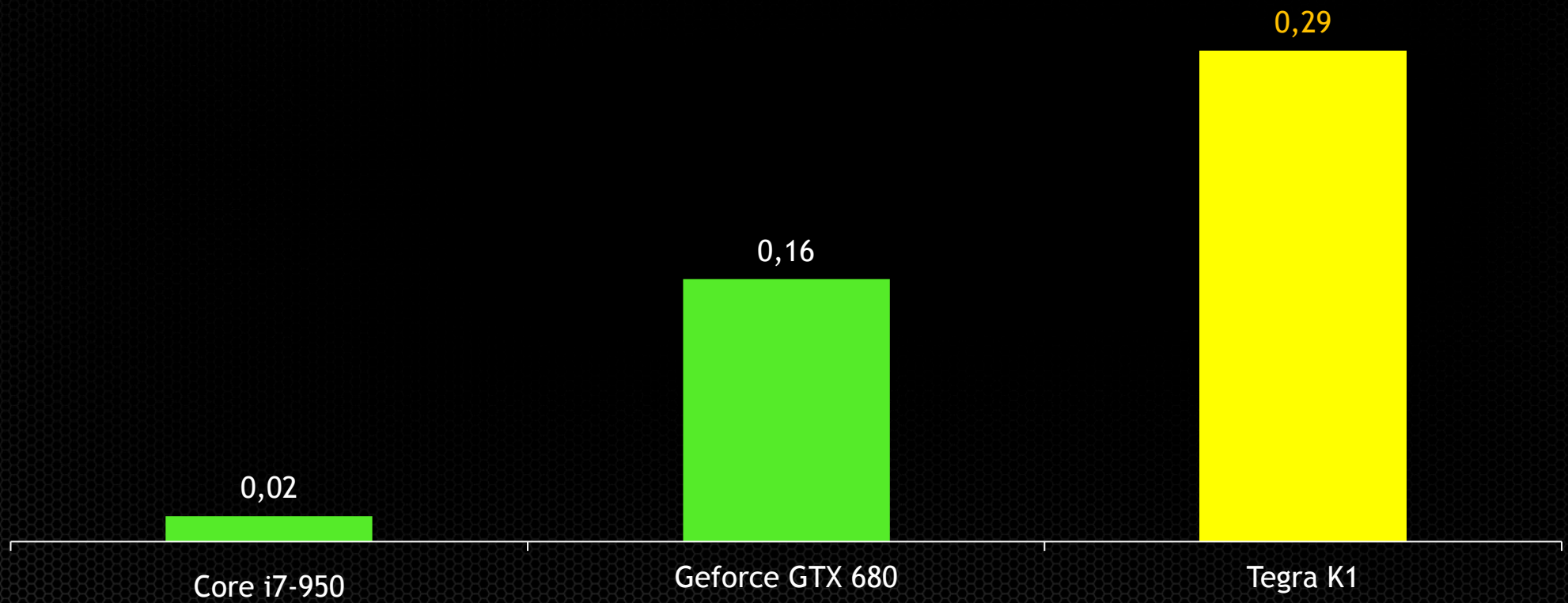
Single Frame Noise Reduction



NVIDIA.

Architectures

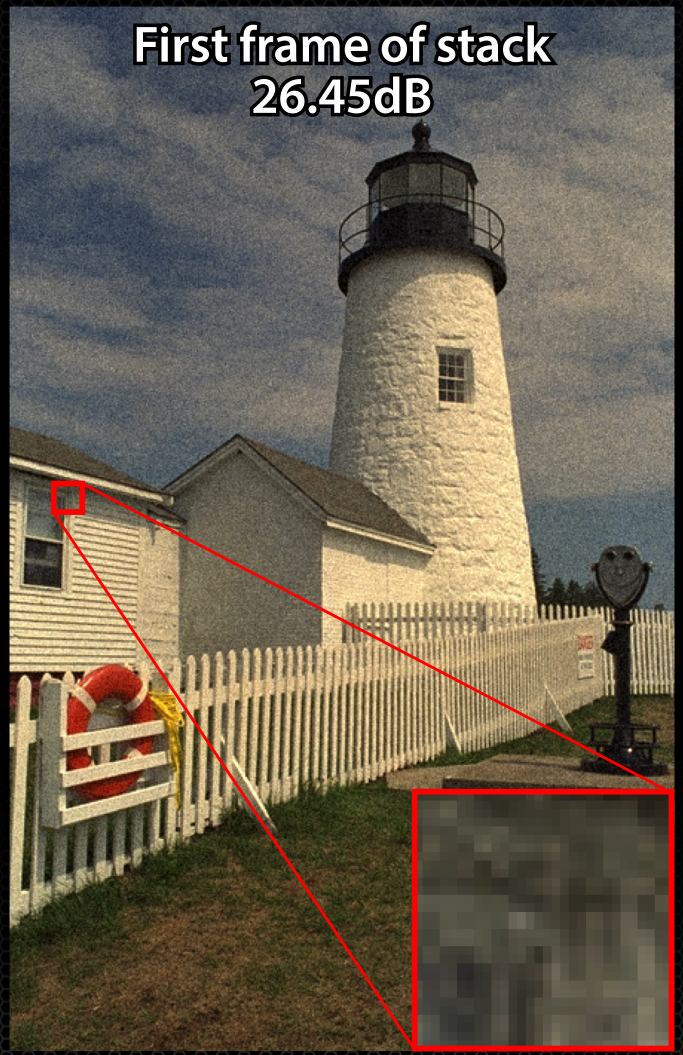
MPix/s/Watt, k=16, 0.25MPix
(the higher the better)



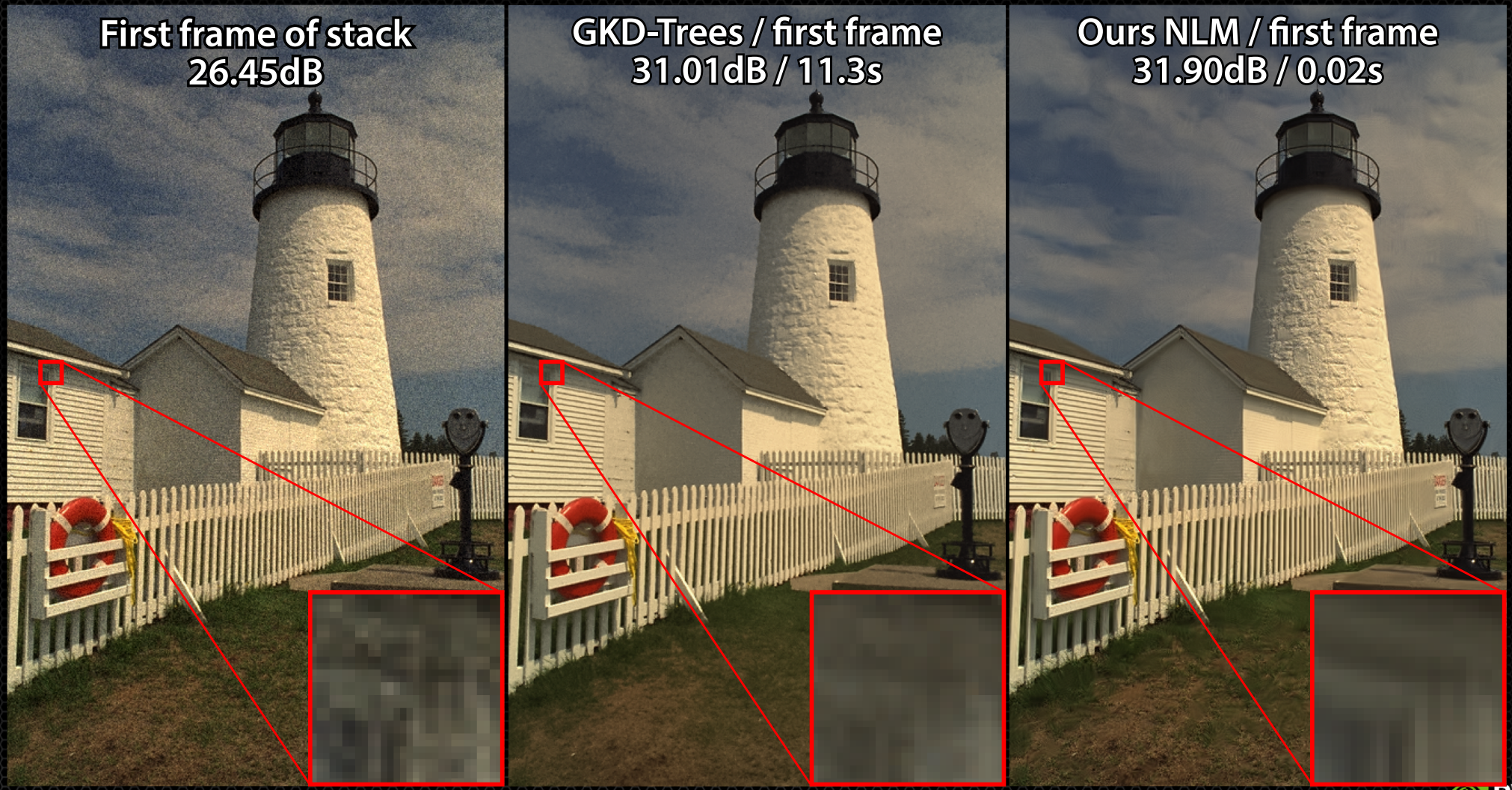
NVIDIA

Burst Denoising - Single Frame

Burst Denoising - Single Frame



Burst Denoising - Single Frame

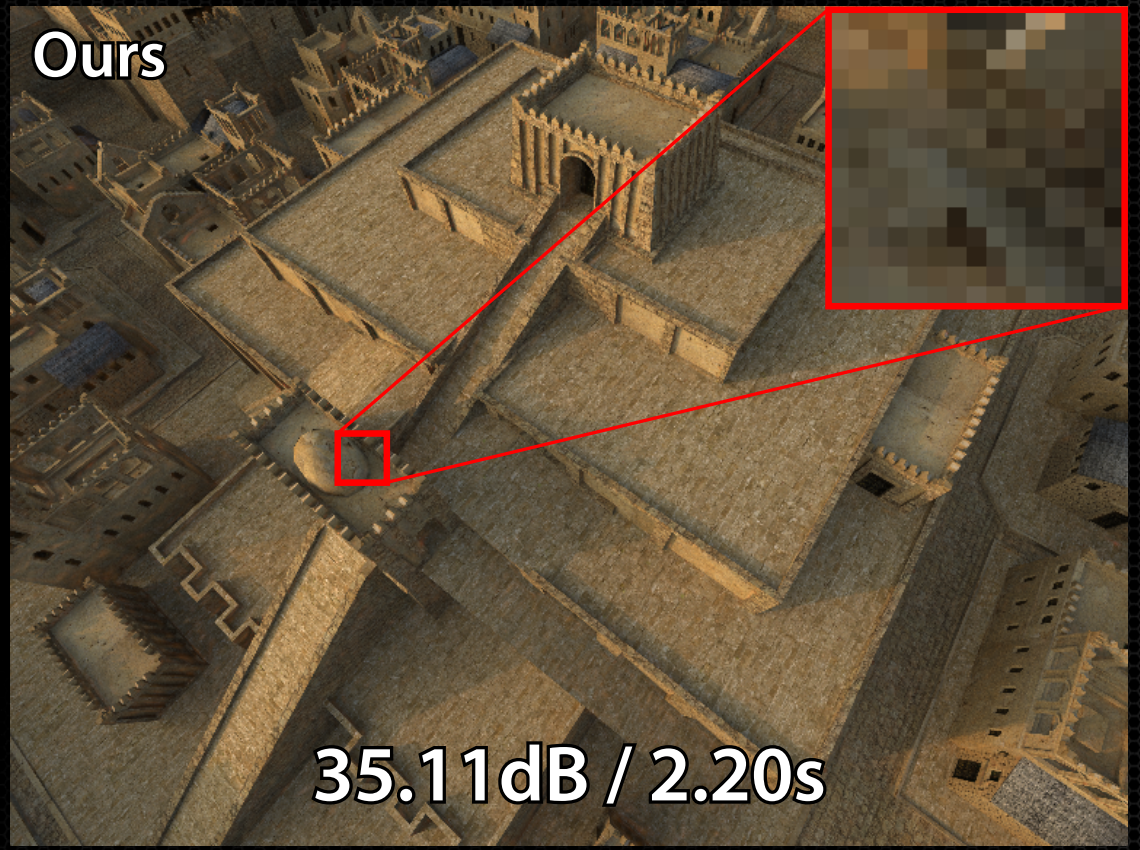


Burst Denoising - All Frames

Burst Denoising - All Frames

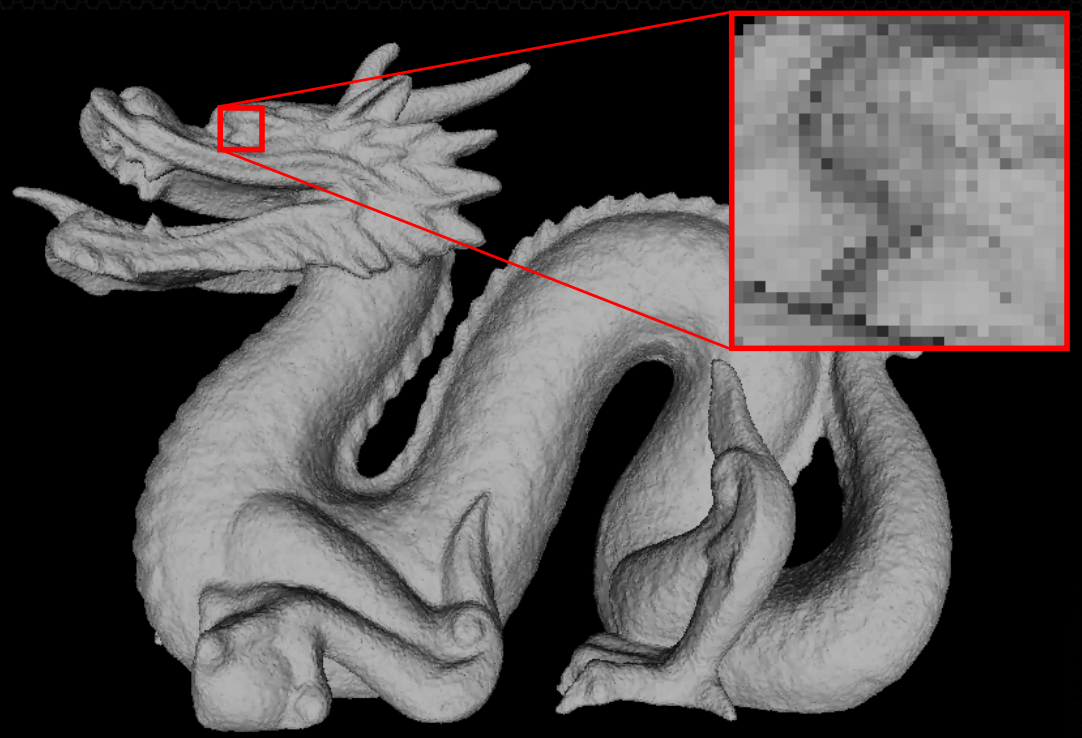
Global Illumination

Global Illumination



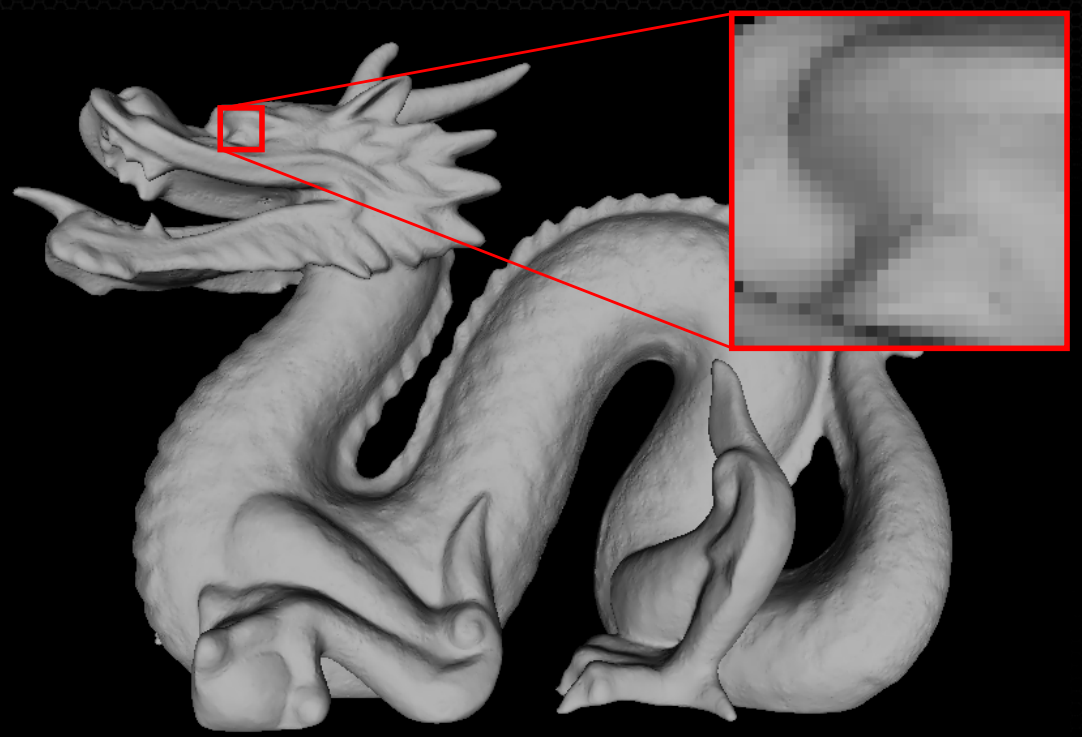
Global Illumination

Geometry Noise Reduction



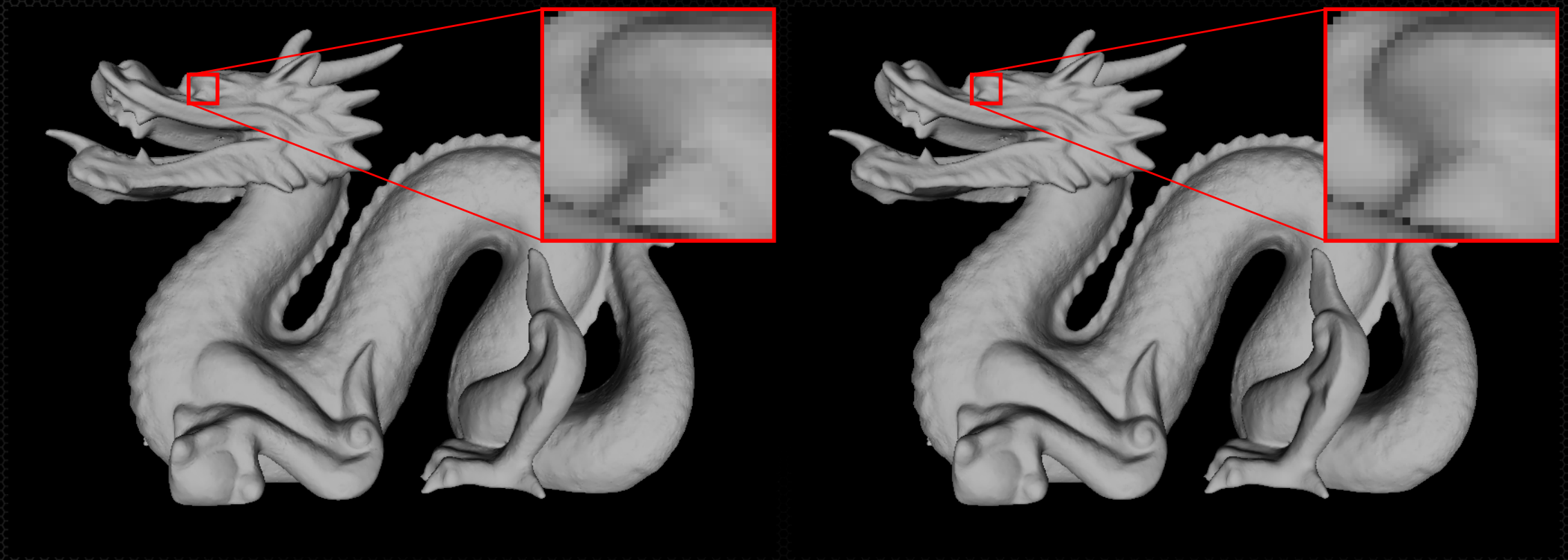
Noisy Input

Geometry Noise Reduction



Ours

Geometry Noise Reduction

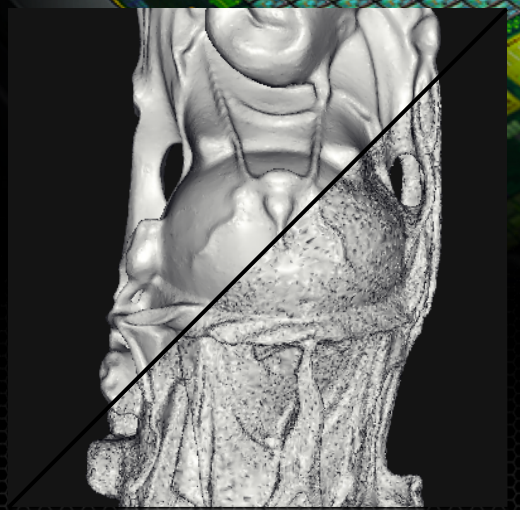


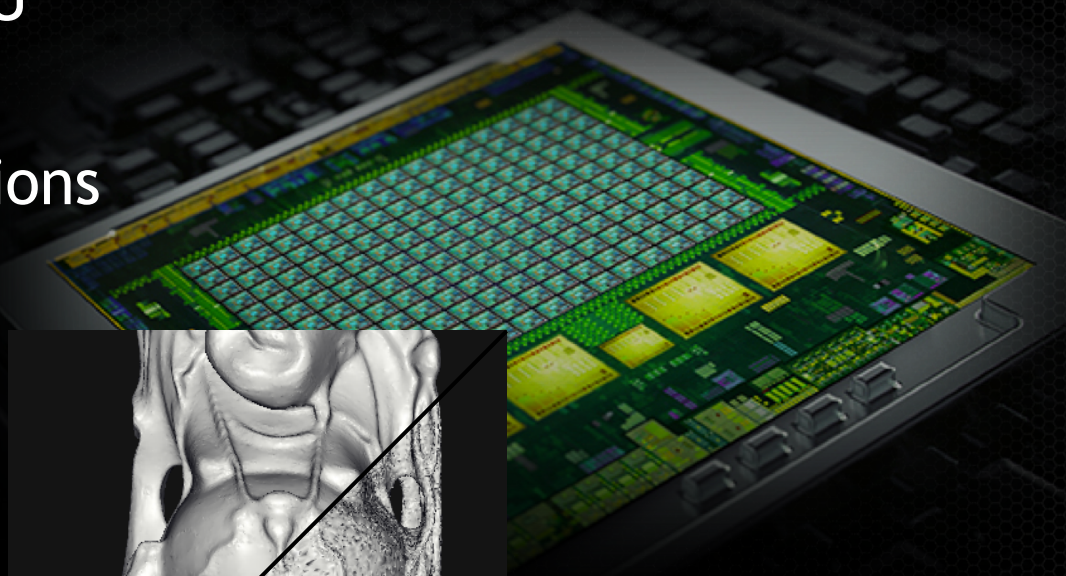
Ours

Exhaustive Search

Conclusions

- Efficient implementation on GPU
- High image quality
- Applicable to different applications





Thank you

- Paper and Binary:
  - http://bit.ly/fast-ann

NVIDIA.