Difference Operator - TECH

AUGUSTO MARTINS – 01656520

2018S - ADVANCED TOPICS IN PARALLEL COMPUTING

UNIVERSITY OF VIENNA

Implementation Steps

1

How to efficiently implement the difference operation?

2

How to parallelize the operation?

3

Loop Unrolling.

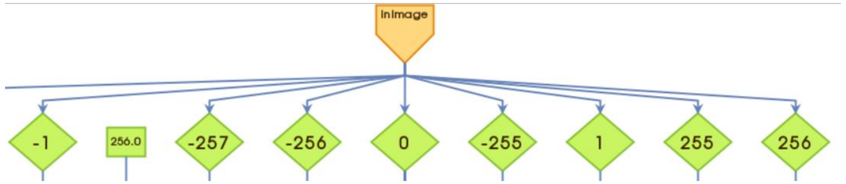
Difference Operator Kernel – Part I

```
DFEVar inImage = io.input("inImage", TYPE);

DFEVar upLeft   = stream.offset(inImage, -(width + 1));
DFEVar up         = stream.offset(inImage, - width);
DFEVar upRight = stream.offset(inImage, -(width - 1));

DFEVar left     = stream.offset(inImage, -1);
DFEVar point = stream.offset(inImage,  0);
DFEVar right    = stream.offset(inImage,  1);

DFEVar downLeft     = stream.offset(inImage, width - 1);
DFEVar down          = stream.offset(inImage, width);
DFEVar downRight = stream.offset(inImage, width + 1);
```



Difference Operator Kernel – Part II

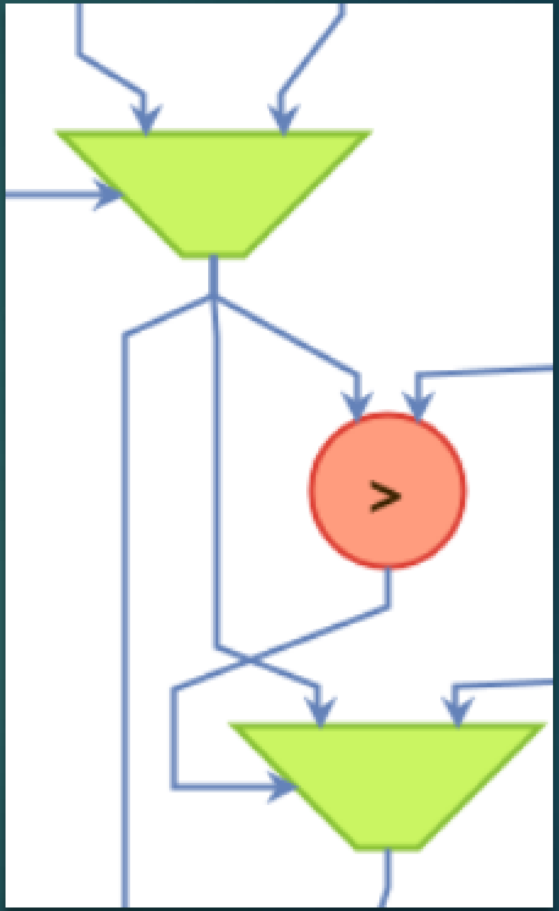
```
DFEVar max = constant.zero(TYPE);

max = (firstLine | firstCol)  ? max : KernelMath.abs(upLeft - point);
max =  firstLine              ? max : KernelMath.max(max, KernelMath.abs(up - point));
max = (firstLine | lastCol)   ? max : KernelMath.max(max, KernelMath.abs(upRight - point));

max = firstCol  ? max : KernelMath.max(max, KernelMath.abs(left - point));
max = lastCol  ? max : KernelMath.max(max, KernelMath.abs(right - point));

max = (lastLine | firstCol) ? max : KernelMath.max(max, KernelMath.abs(downLeft - point));
max =  lastLine     ? max : KernelMath.max(max, KernelMath.abs(down - point));
max = (lastLine | lastCol)  ? max : KernelMath.max(max, KernelMath.abs(downRight - point));

io.output("outImage", max, TYPE);
```



Calling the Difference Operator Kernel from the Host

```
int main(void) {
  printf("Loading image.\n");
  int32_t *inImage;
  int width = 0, height = 0;
  loadImage("lena.ppm", &inImage, &width, &height, 1);

  uint64_t n = width * height;
  size_t size = n * sizeof(int32_t);
  int32_t *outImage = malloc(size);

  printf("Running Kernel.\n");
  DifferenceOperator(n, inImage, size, outImage, size);

  printf("Saving image.\n");
  writeImage("lena_difference.ppm", outImage, width, height, 1);
  return 0;
}
```