High-Performance Programming and Execution of a Coral Biodiversity Mapping Algorithm Using Chapel

Scott Bachman
National Center for Atmospheric Research
Hewlett Packard Enterprise

Rebecca Green and Helen Fox
The Coral Reef Alliance

Anna Bakker and Sam Purkis
University of Miami

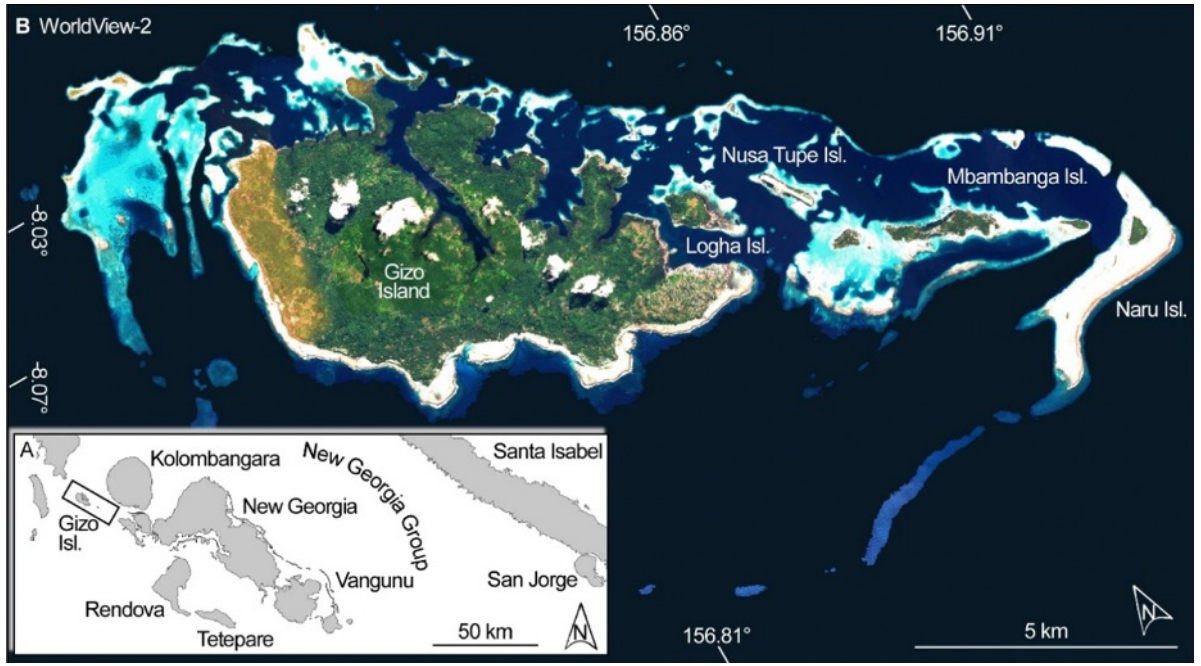
Ben Harshbarger
Hewlett Packard Enterprise

CORAL REEF BIODIVERSITY (IMAGE ANALYSIS)

Background

- **Problem:** Coral reefs are widespread, but small and hard to reach. Surveying and measuring is difficult, time-consuming, and expensive.

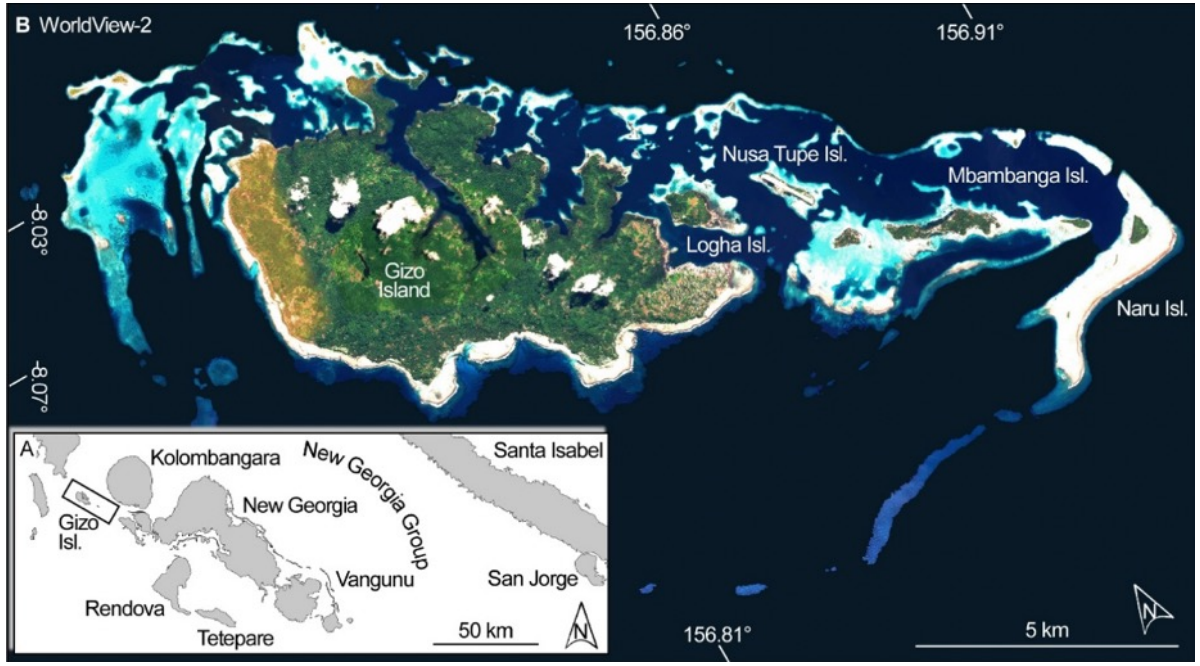
Do corals and fish correspond to seascape patterns in the reef as seen by satellite?



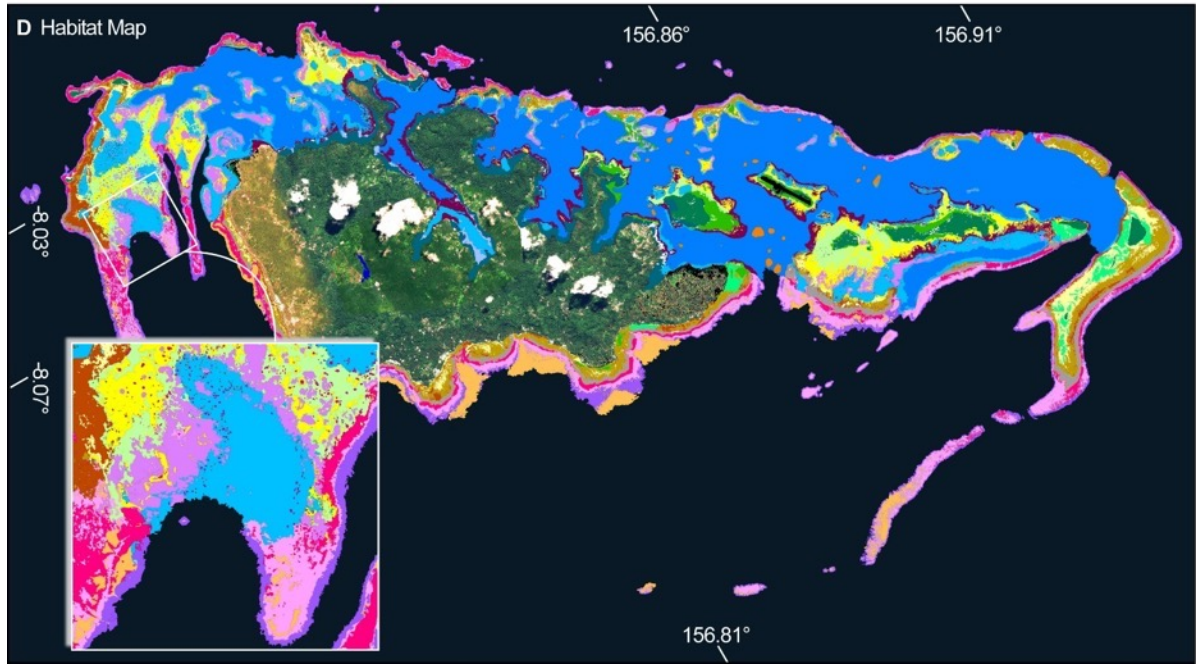
CORAL REEF BIODIVERSITY (IMAGE ANALYSIS)

Background

Satellite image

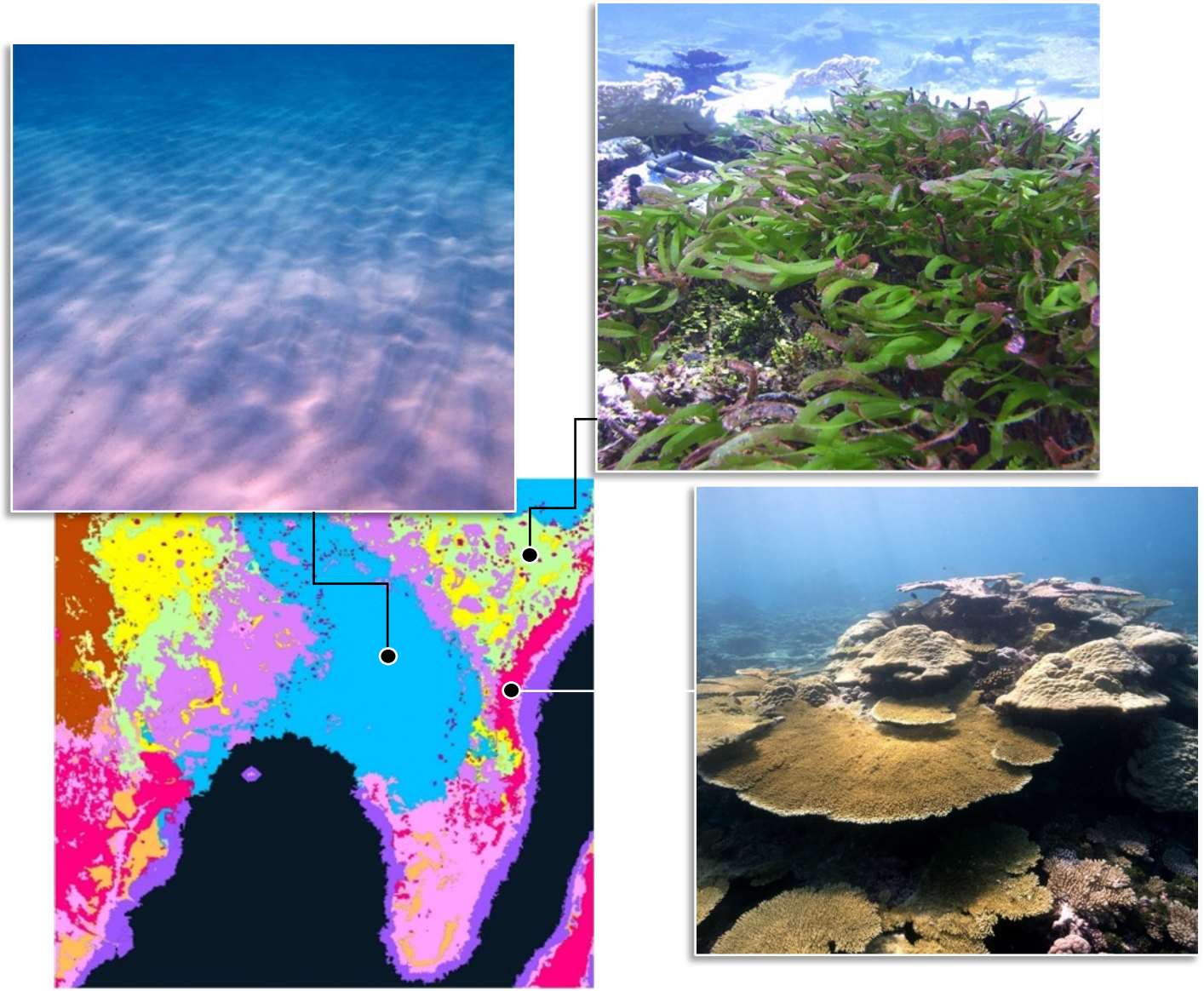


Rasterized habitat map



CORAL REEF BIODIVERSITY (IMAGE ANALYSIS)

Background

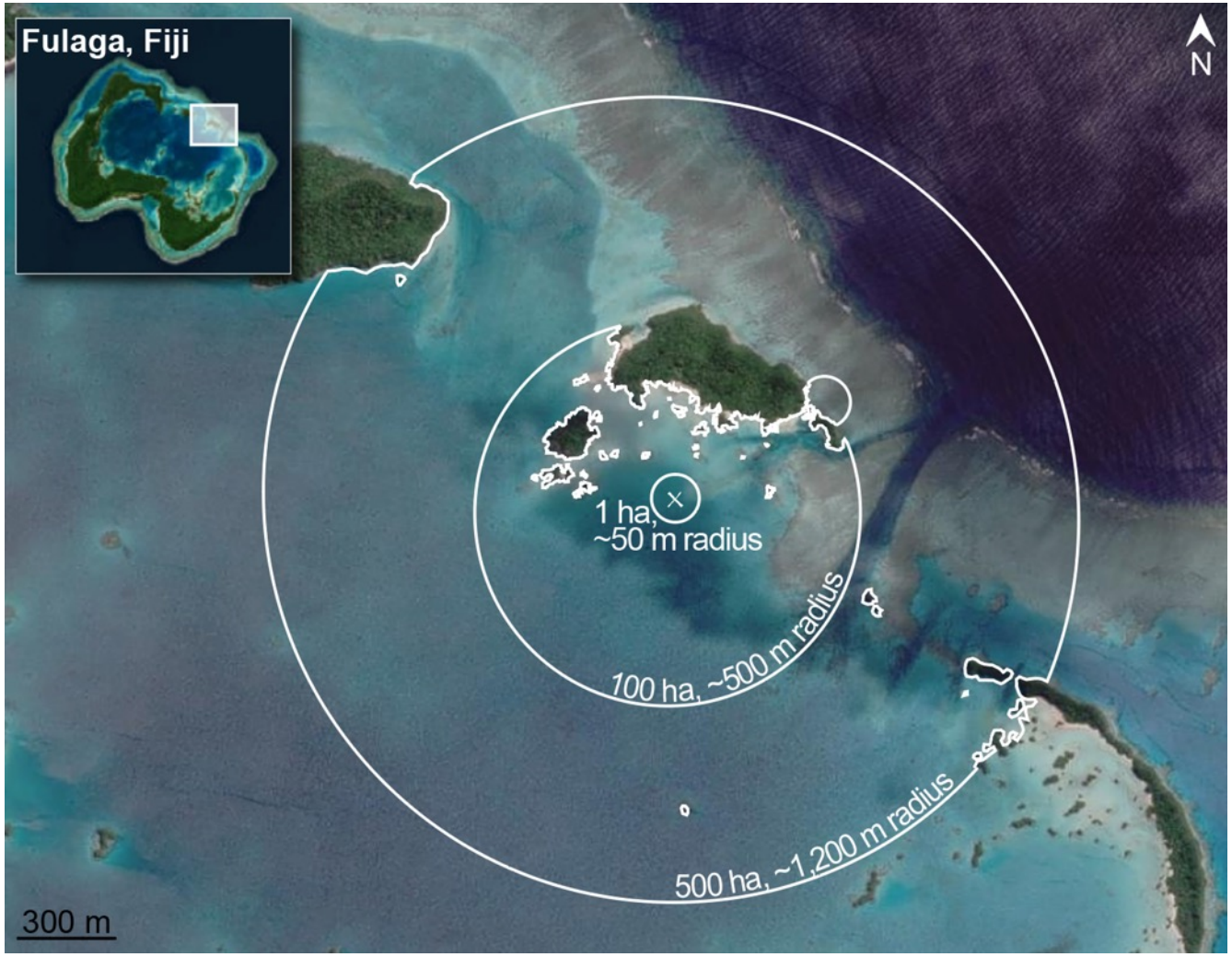


CORAL REEF BIODIVERSITY (IMAGE ANALYSIS)

Background

Challenge: How much habitat diversity occurs in a circle of size X?

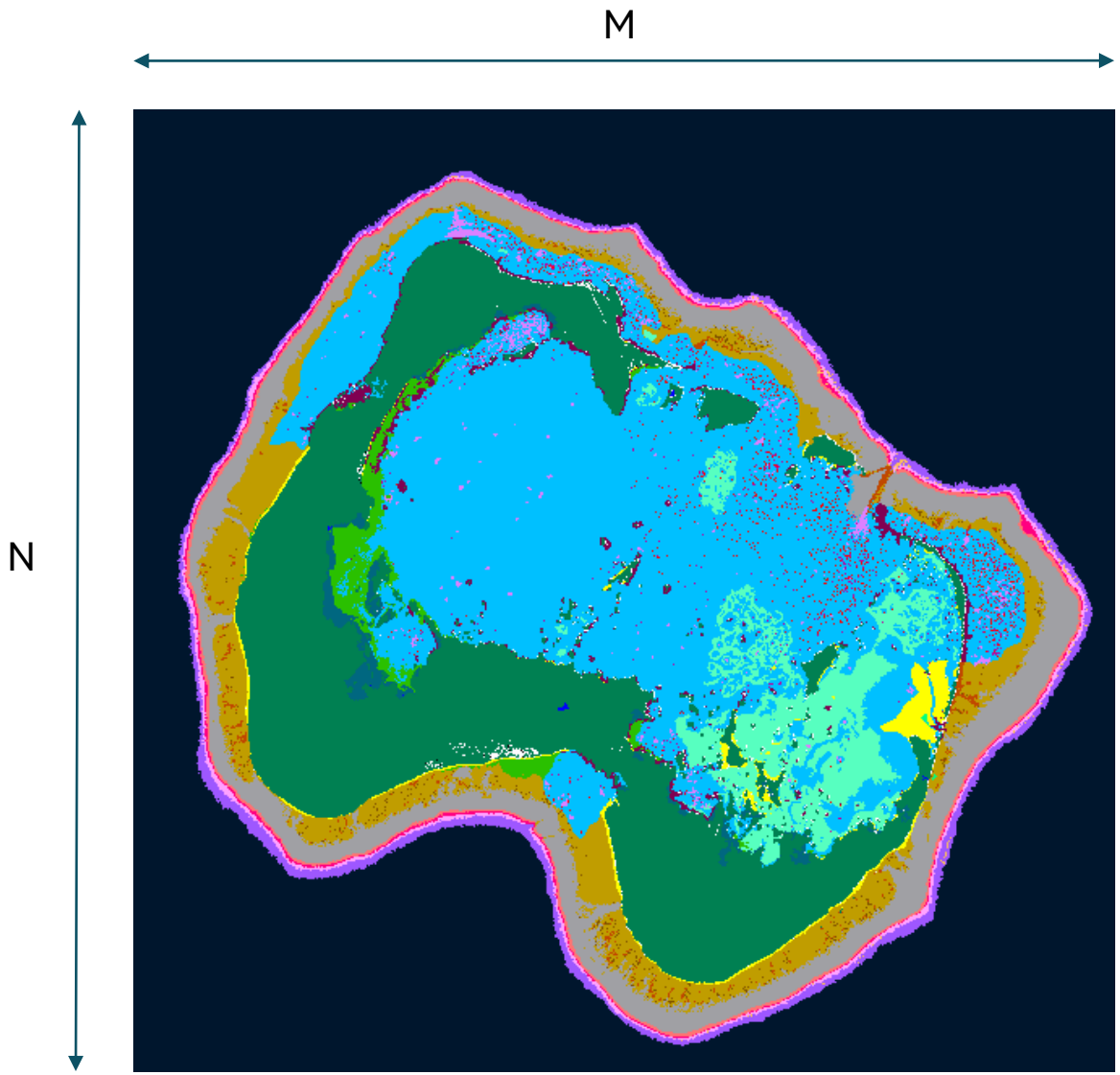
Need to perform this calculation for each point in the image.



CORAL REEF BIODIVERSITY (IMAGE ANALYSIS)

This Effort

1. Read in a (M x N) raster image of habitat data

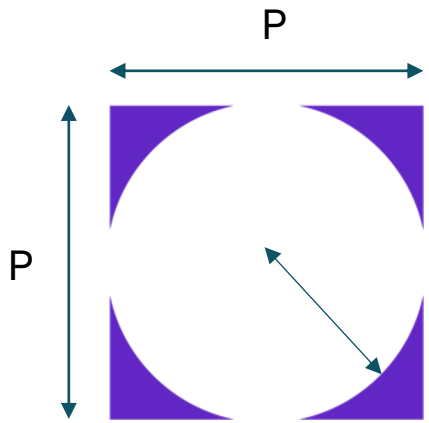


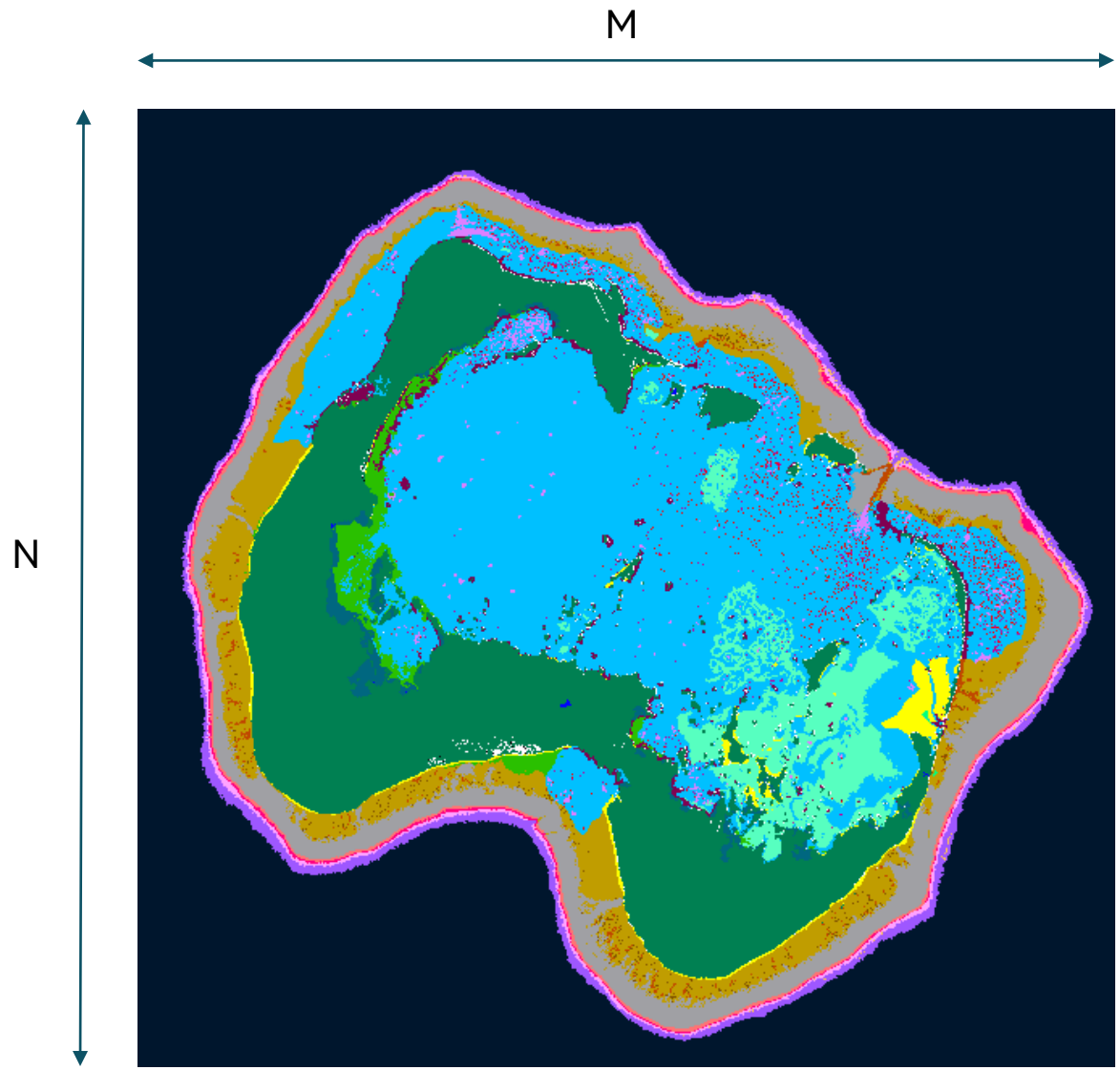
CORAL REEF BIODIVERSITY (IMAGE ANALYSIS)

This Effort

1. Read in a (M x N) raster image of habitat data

2. Create a (P x P) mask to find all points within a given radius.





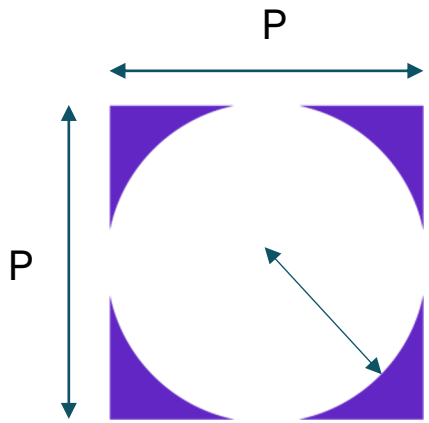
CORAL REEF BIODIVERSITY (IMAGE ANALYSIS)

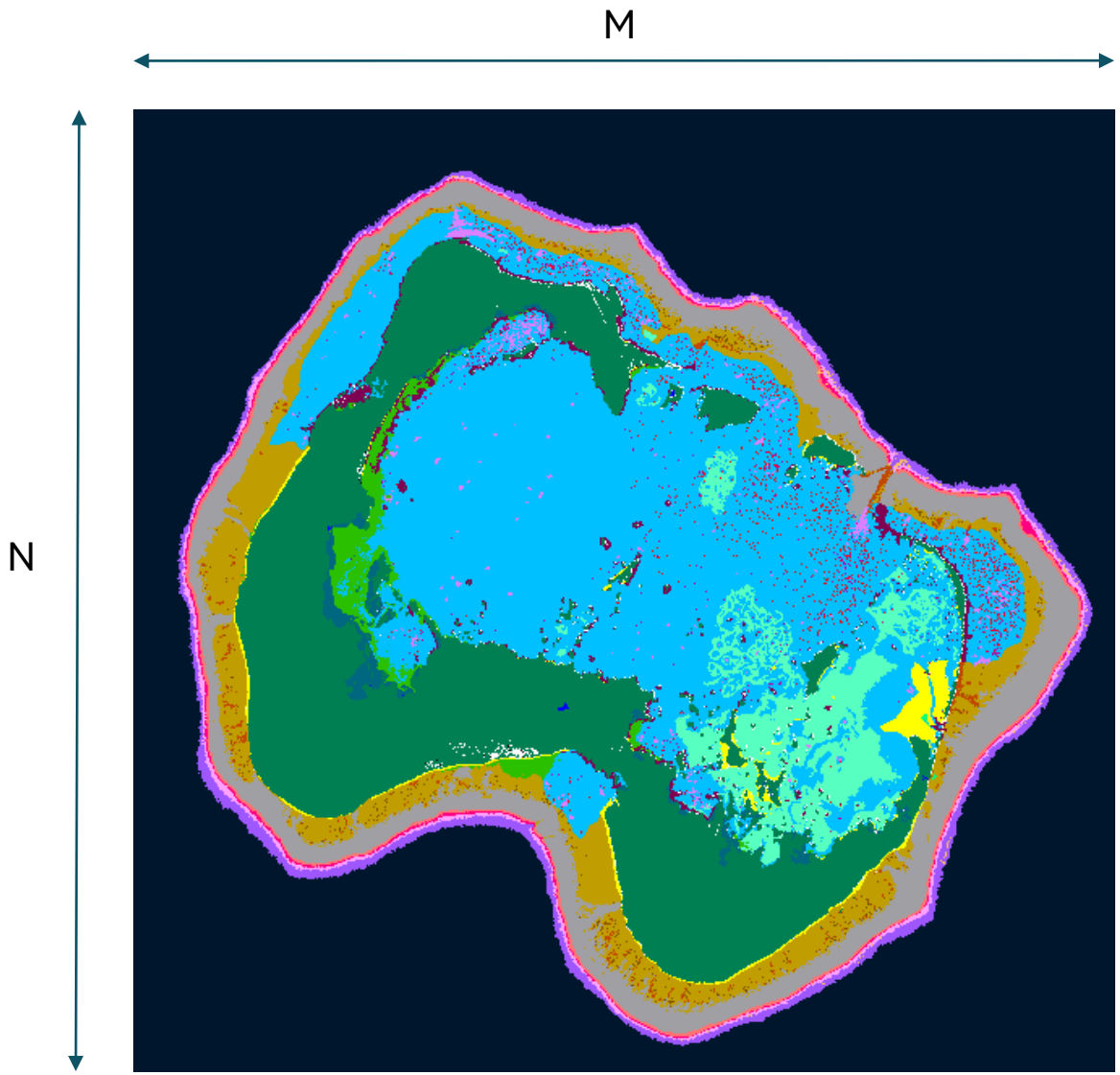
This Effort

1. Read in a (M x N) raster image of habitat data

2. Create a (P x P) mask to find all points within a given radius.

3. Convolve this mask over the entire domain and perform a weighted reduce at each location.





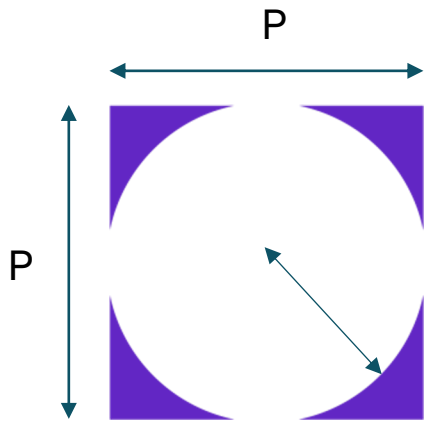
CORAL REEF BIODIVERSITY (IMAGE ANALYSIS)

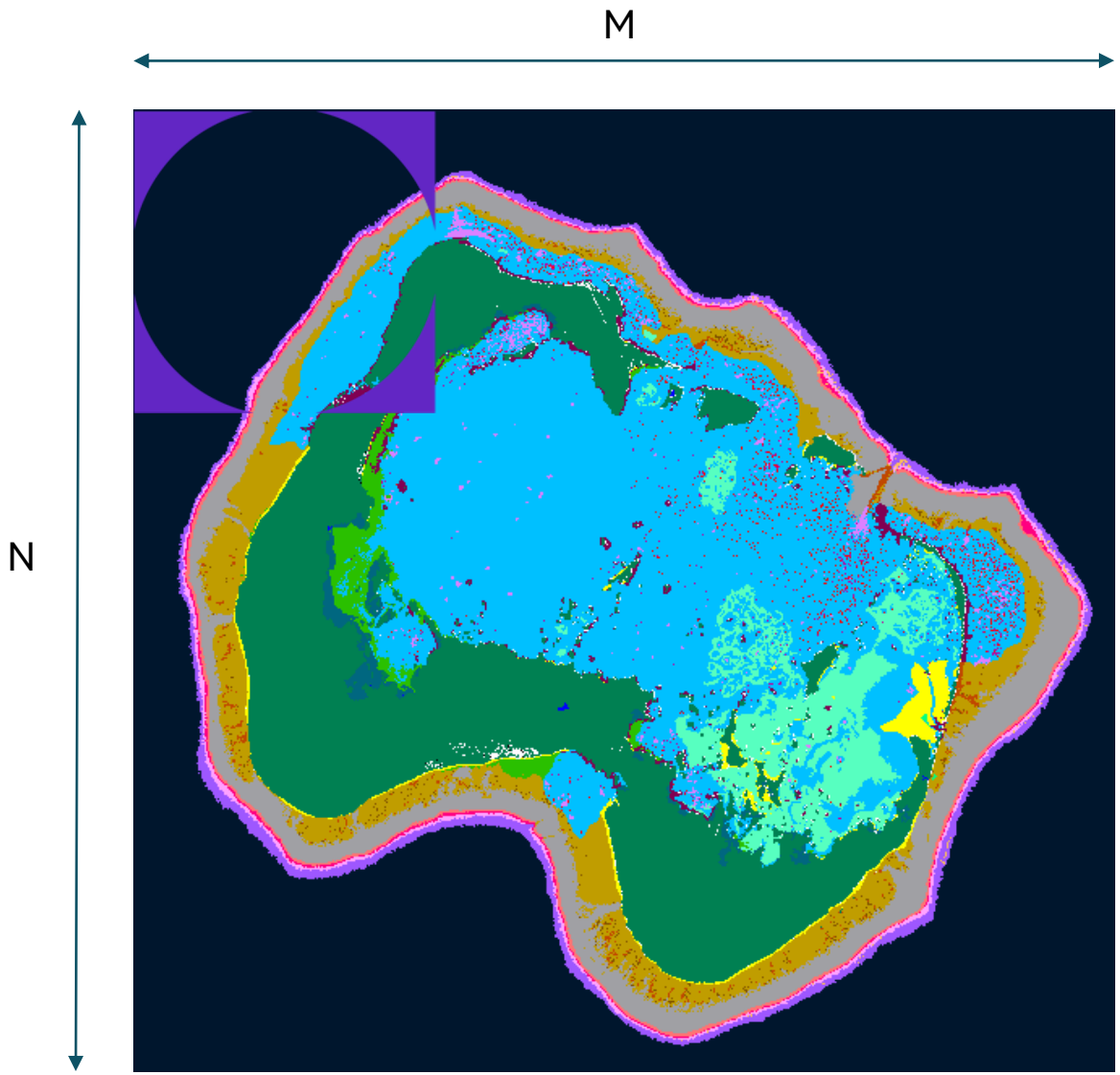
This Effort

1. Read in a (M x N) raster image of habitat data

2. Create a (P x P) mask to find all points within a given radius.

3. Convolve this mask over the entire domain and perform a weighted reduce at each location.





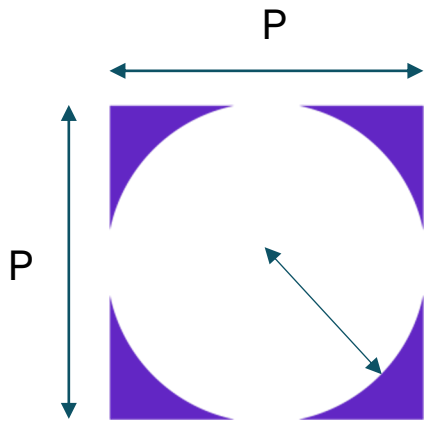
CORAL REEF BIODIVERSITY (IMAGE ANALYSIS)

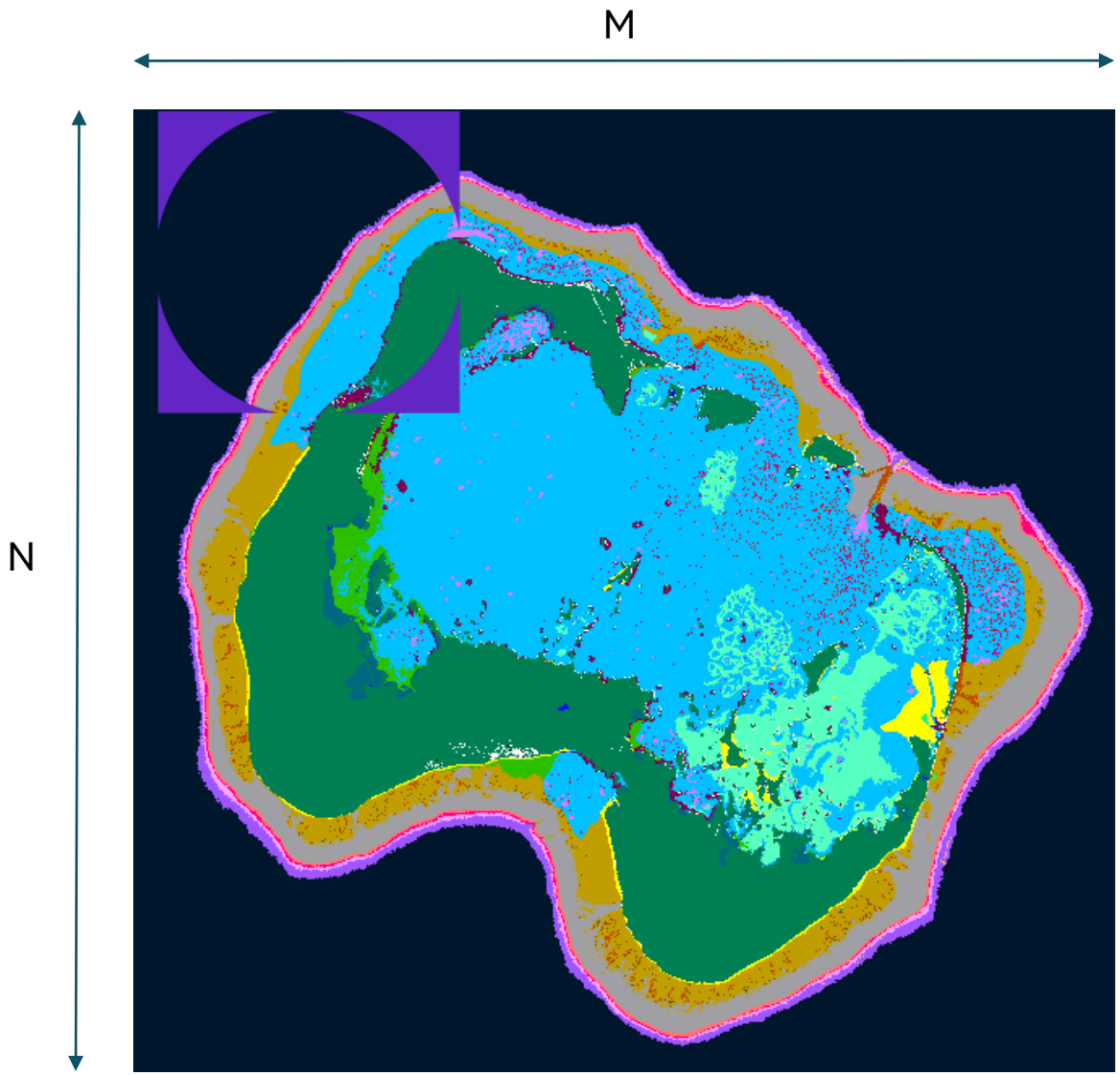
This Effort

1. Read in a (M x N) raster image of habitat data

2. Create a (P x P) mask to find all points within a given radius.

3. Convolve this mask over the entire domain and perform a weighted reduce at each location.





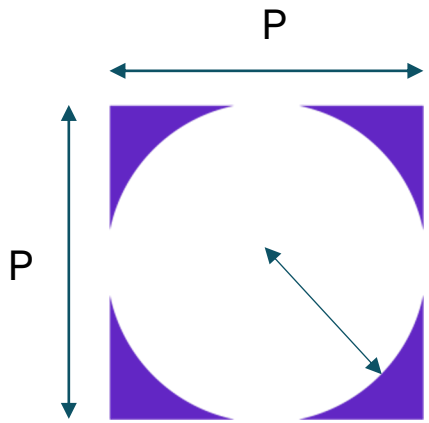
CORAL REEF BIODIVERSITY (IMAGE ANALYSIS)

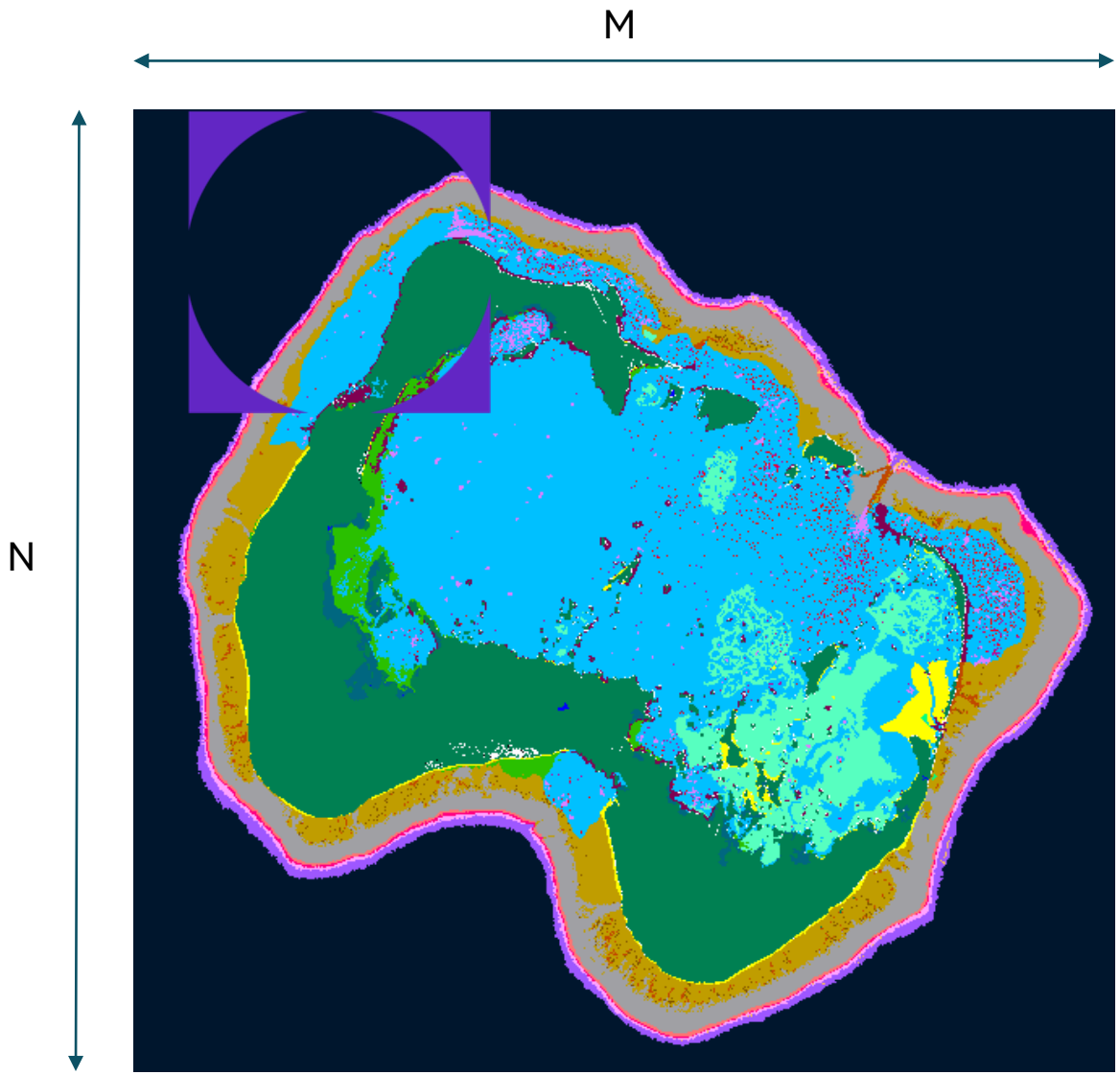
This Effort

1. Read in a (M x N) raster image of habitat data

2. Create a (P x P) mask to find all points within a given radius.

3. Convolve this mask over the entire domain and perform a weighted reduce at each location.





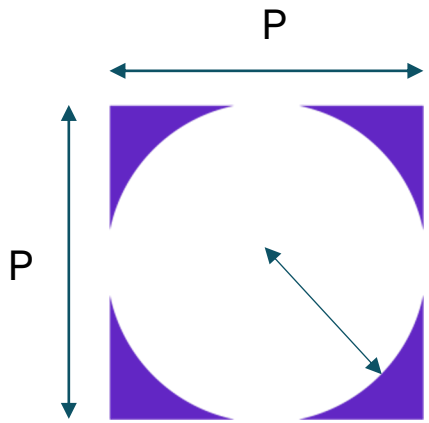
CORAL REEF BIODIVERSITY (IMAGE ANALYSIS)

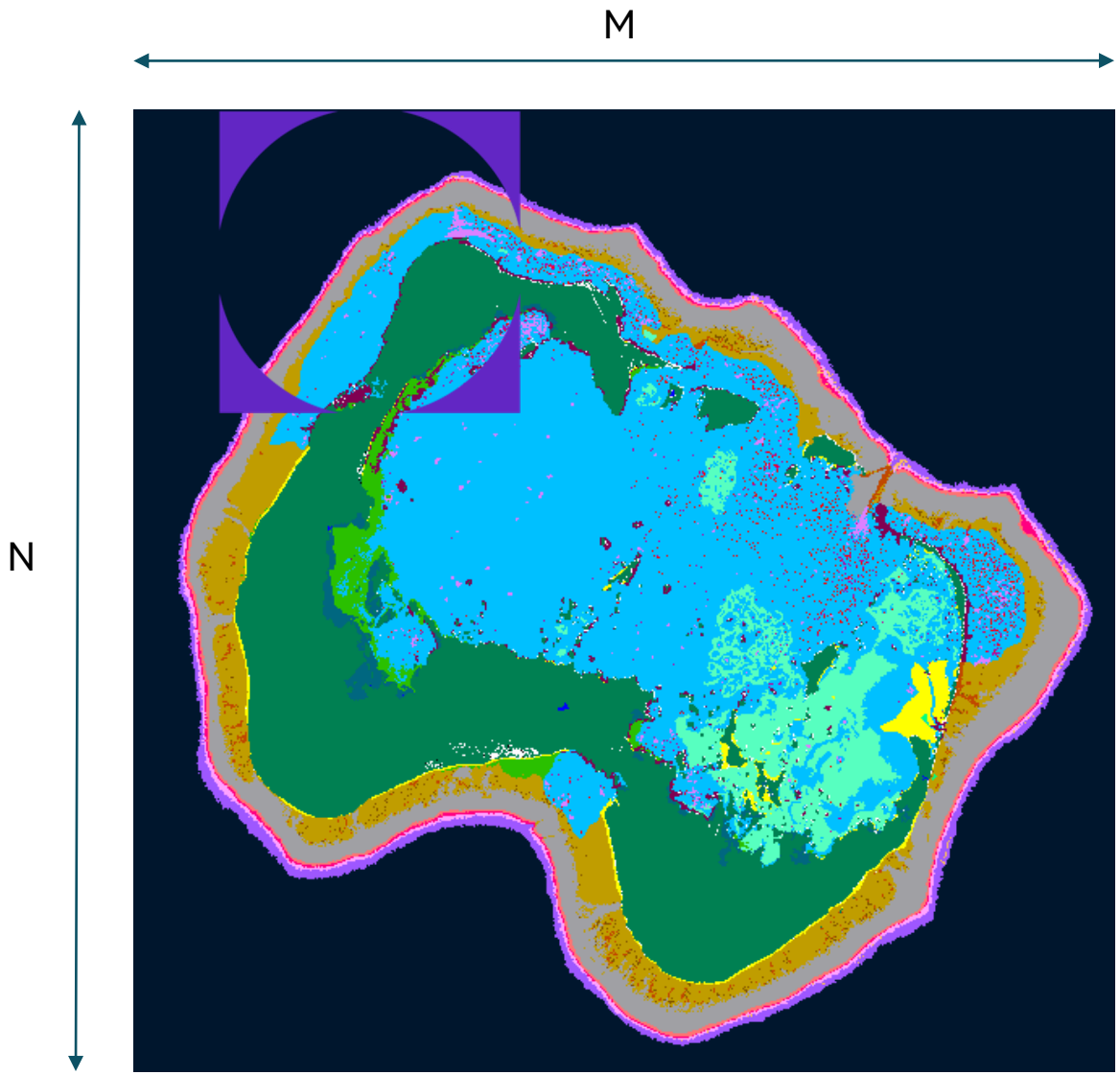
This Effort

1. Read in a (M x N) raster image of habitat data

2. Create a (P x P) mask to find all points within a given radius.

3. Convolve this mask over the entire domain and perform a weighted reduce at each location.





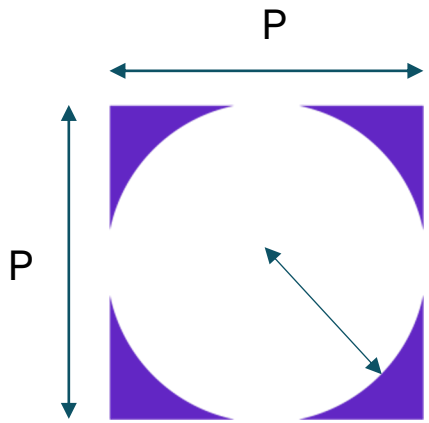
CORAL REEF BIODIVERSITY (IMAGE ANALYSIS)

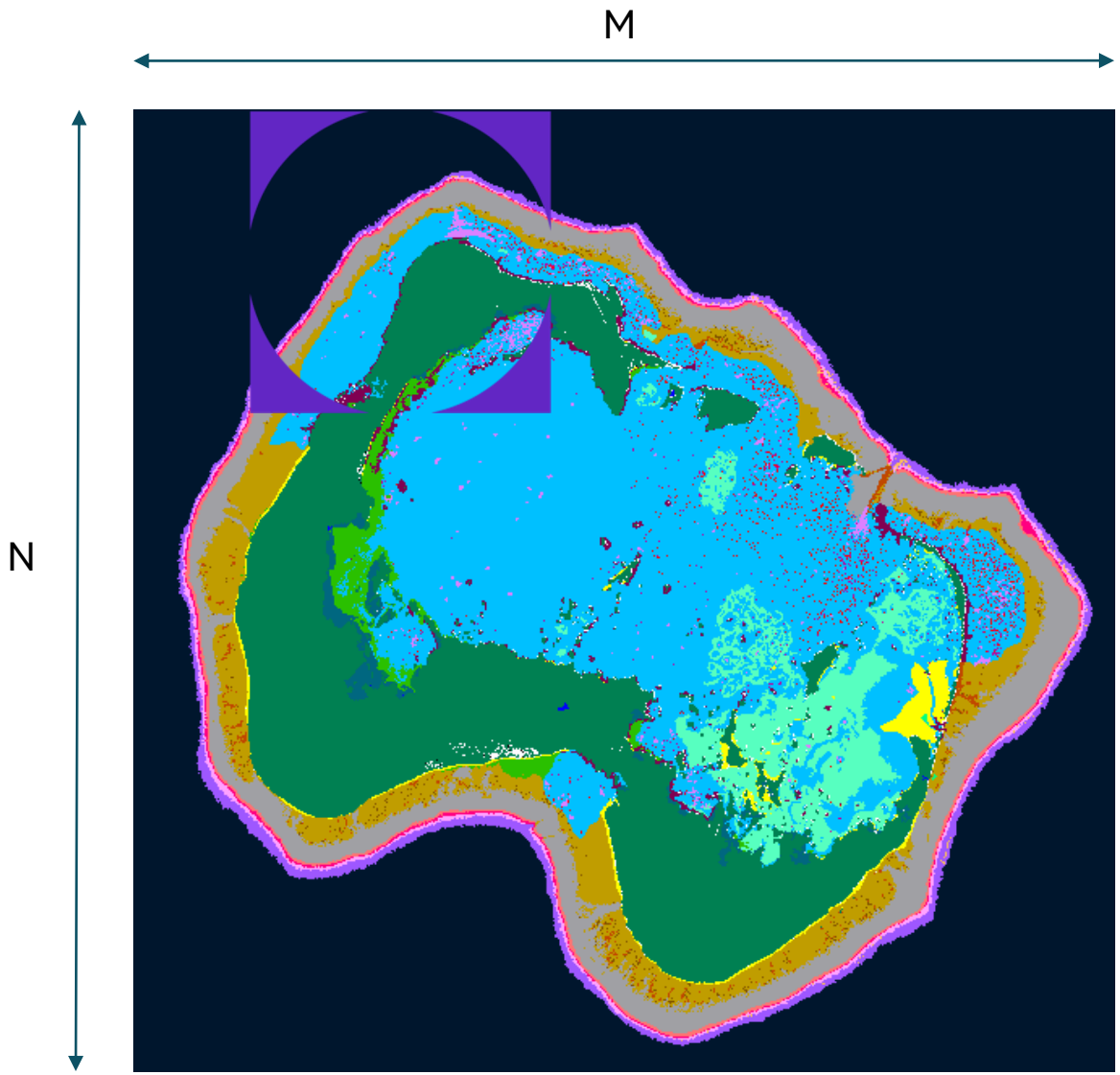
This Effort

1. Read in a (M x N) raster image of habitat data

2. Create a (P x P) mask to find all points within a given radius.

3. Convolve this mask over the entire domain and perform a weighted reduce at each location.





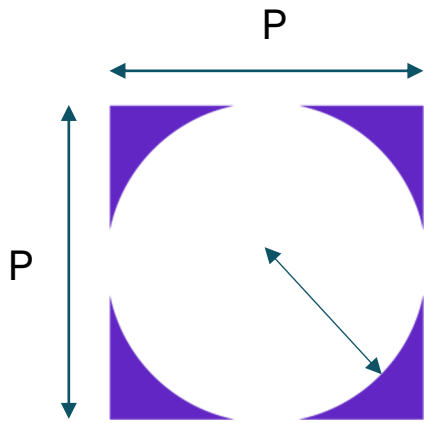
CORAL REEF BIODIVERSITY (IMAGE ANALYSIS)

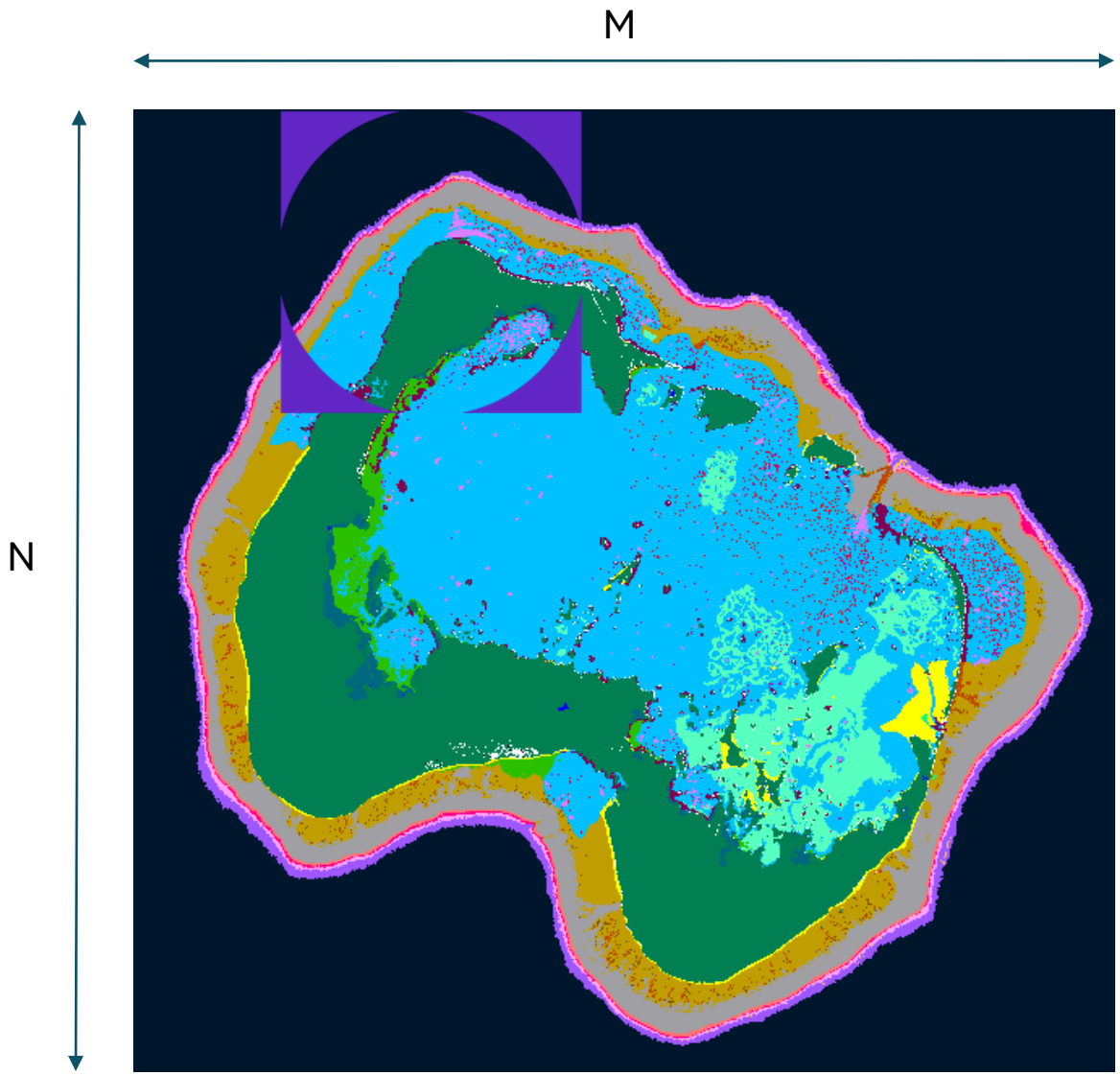
This Effort

1. Read in a (M x N) raster image of habitat data

2. Create a (P x P) mask to find all points within a given radius.

3. Convolve this mask over the entire domain and perform a weighted reduce at each location.



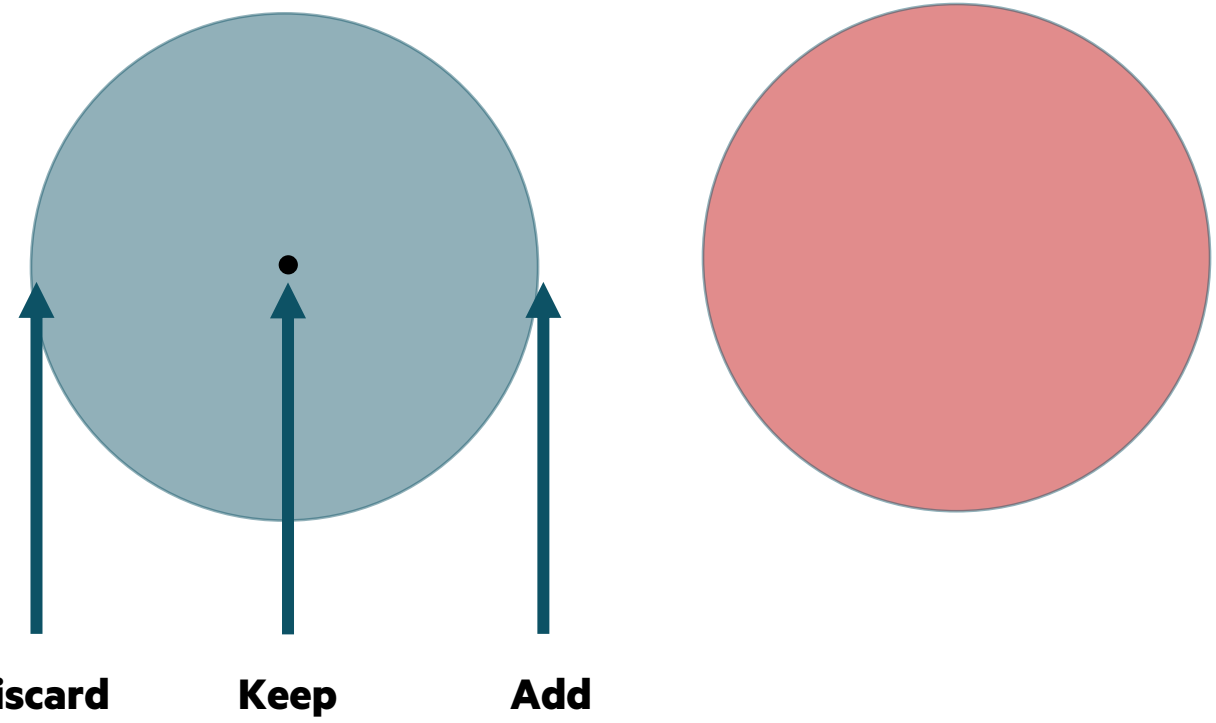


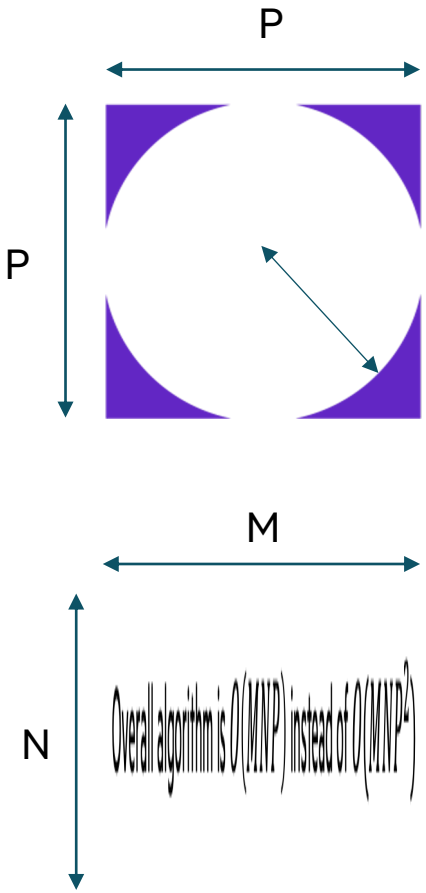
CORAL REEF BIODIVERSITY (IMAGE ANALYSIS)

The code

The basic algorithm essentially involves calculating a histogram at each point.

"How many habitats of each type are in the neighborhood?"





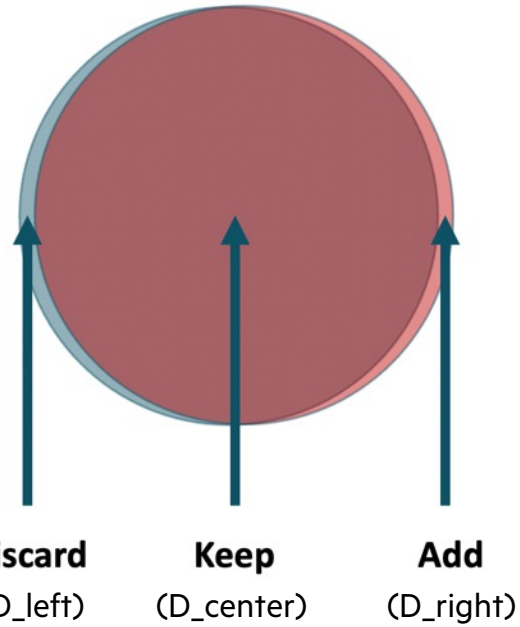
- Saves roughly a factor of P calculations.

Overall algorithm is $O(MNP)$ instead of $O(MNP^2)$

CORAL REEF BIODIVERSITY (IMAGE ANALYSIS)

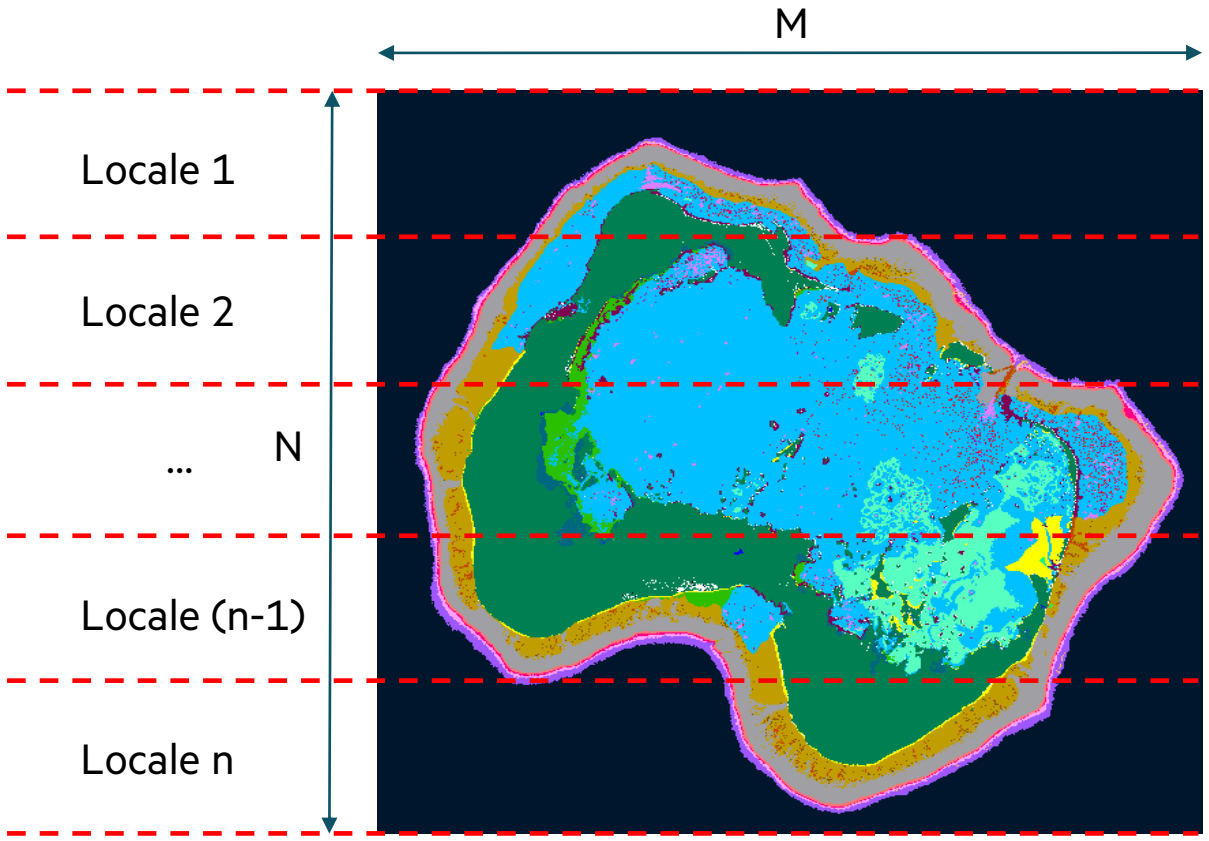
The code

```
proc create_distance_mask(radius : real, dx : real, nx : int) {

  const D : domain(2, int) = {-nx..nx, -nx..nx};
  var D_center : sparse subdomain(D);
  var D_left : sparse subdomain(D);
  var D_right : sparse subdomain(D);

  var dist : [D] real;

  var center_mask : [D_center] bool;
  var left_mask : [D_left] bool;
  var right_mask : [D_right] bool;



                ...



  // Define center mask.
  for (i,j) in dist.domain do {
    dist[i,j] = dx * sqrt(i**2 + j**2);
    if (dist[i,j] < radius) {
      D_center += (i,j);
    }
  }
```



CORAL REEF BIODIVERSITY (IMAGE ANALYSIS)

The code




```
// Create Block distribution of interior of PNG
const offset = nx;
const Inner = ImageSpace.expand(-offset);
const myTargetLocales = reshape(Locales, {1..Locales.size, 1..1});
const D = Inner dmapped Block(Inner, targetLocales=myTargetLocales);
var OutputArray : [D] real;
```

CORAL REEF BIODIVERSITY (IMAGE ANALYSIS)

The code

```
coforall loc in Locales do on loc {

  const locImageDomain = Image.domain;
  const locImage : [locImageDomain] Image.eltType = Image;

  const locLeftMaskDomain = LeftMask.domain;
  const locCenterMaskDomain = CenterMask.domain;
  const locRightMaskDomain = RightMask.domain;

                          ...

  // If we are on a reef point, calculate beta diversity

    var num_habitat_pixels = (+ reduce B[1..(d_size-2)]) : real;
    var habitat_frac = num_habitat_pixels / Mask_Size;

    var P = B / num_habitat_pixels;

    var beta = + reduce (dissimilarity * outer(P,P));
    Output[center,point] = habitat_frac * beta + eps;
```

$$B_d = \alpha \sum_i \sum_j (D_{ij} P_i P_j)$$

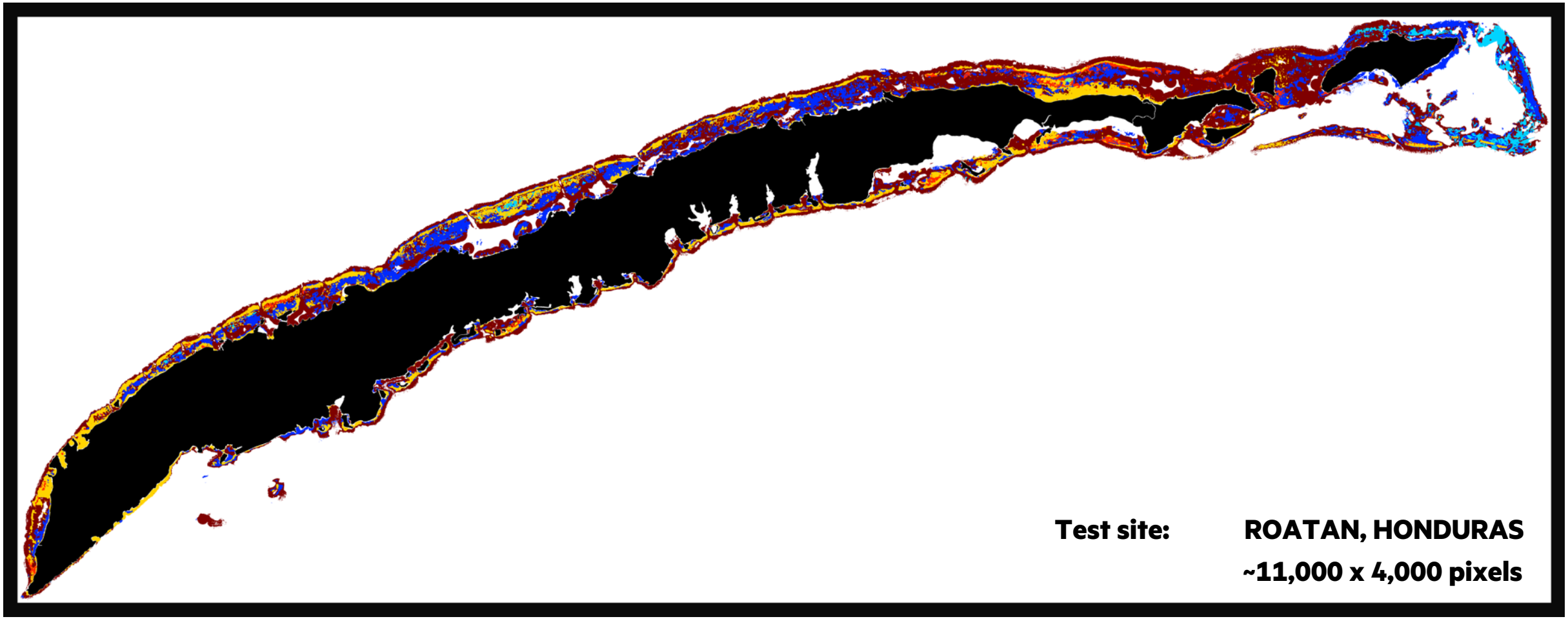
α - fraction of window covered by valid habitat pixels

D_{ij} - dissimilarity coefficient

P_i - fraction of points consisting of habitat *i*

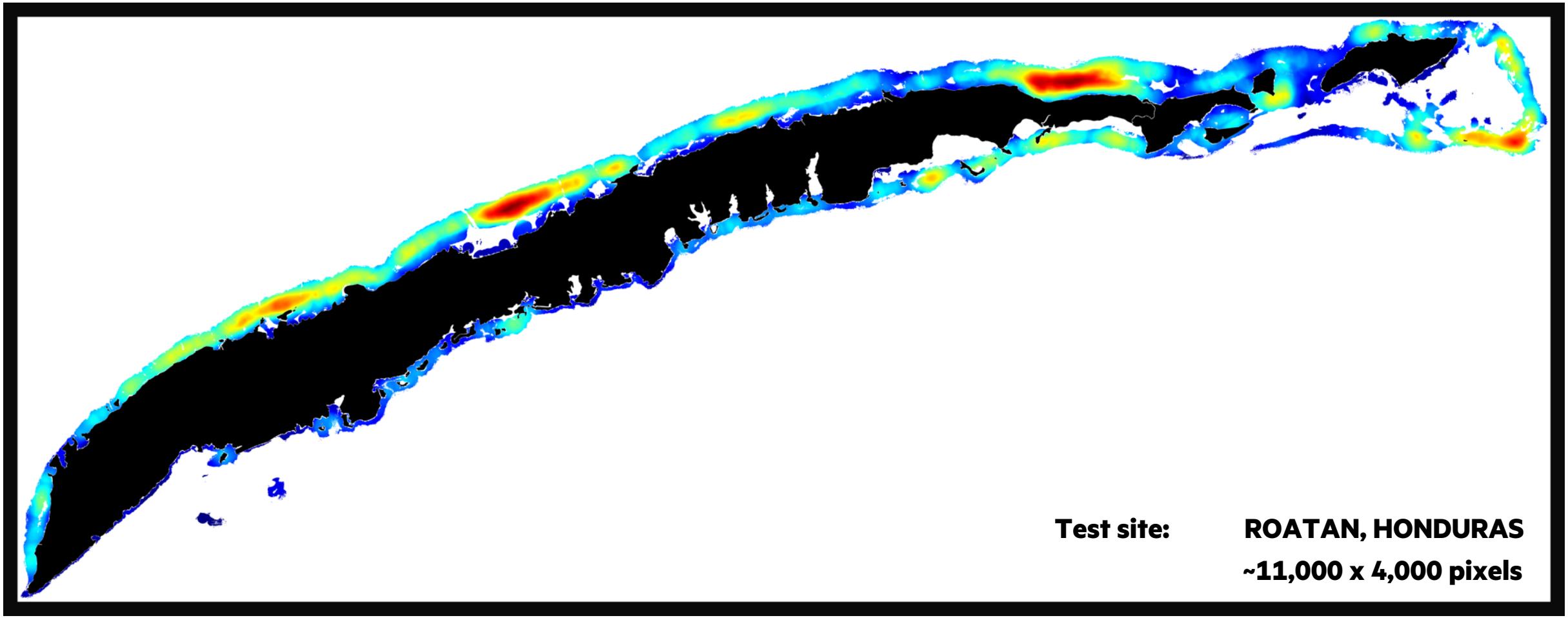
CORAL REEF BIODIVERSITY (IMAGE ANALYSIS)

Performance



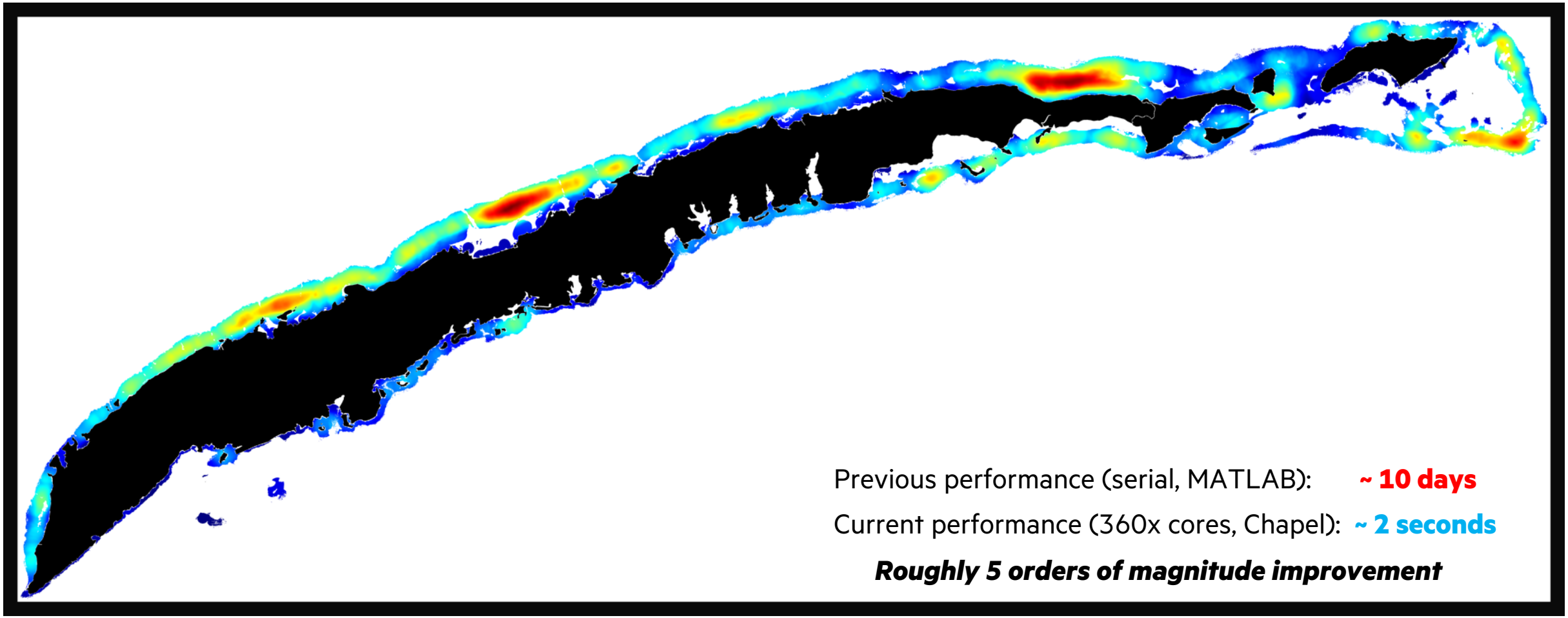
CORAL REEF BIODIVERSITY (IMAGE ANALYSIS)

Performance



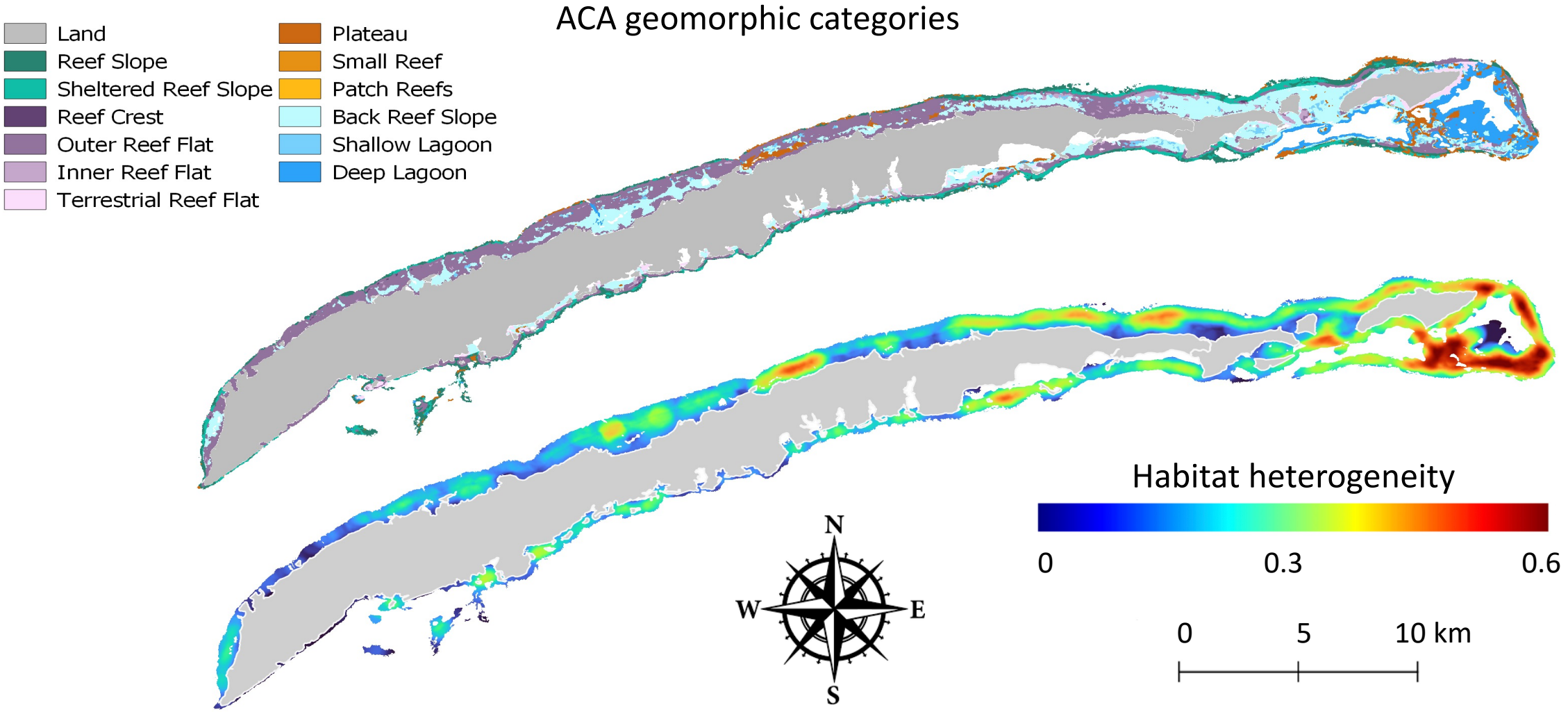
CORAL REEF BIODIVERSITY (IMAGE ANALYSIS)

Performance



- Collaborations/discussions w/ U. Miami, Coral Reef Alliance, U. Leeds (UK), U. Hawaii
- Unexpected shoutout here(!): https://www.youtube.com/watch?v=tlp2_DI6NaI

Performance



Previous performance (serial, MATLAB): **~ Multiple days**

Current performance (360x cores, Chapel): **~ 2 seconds**

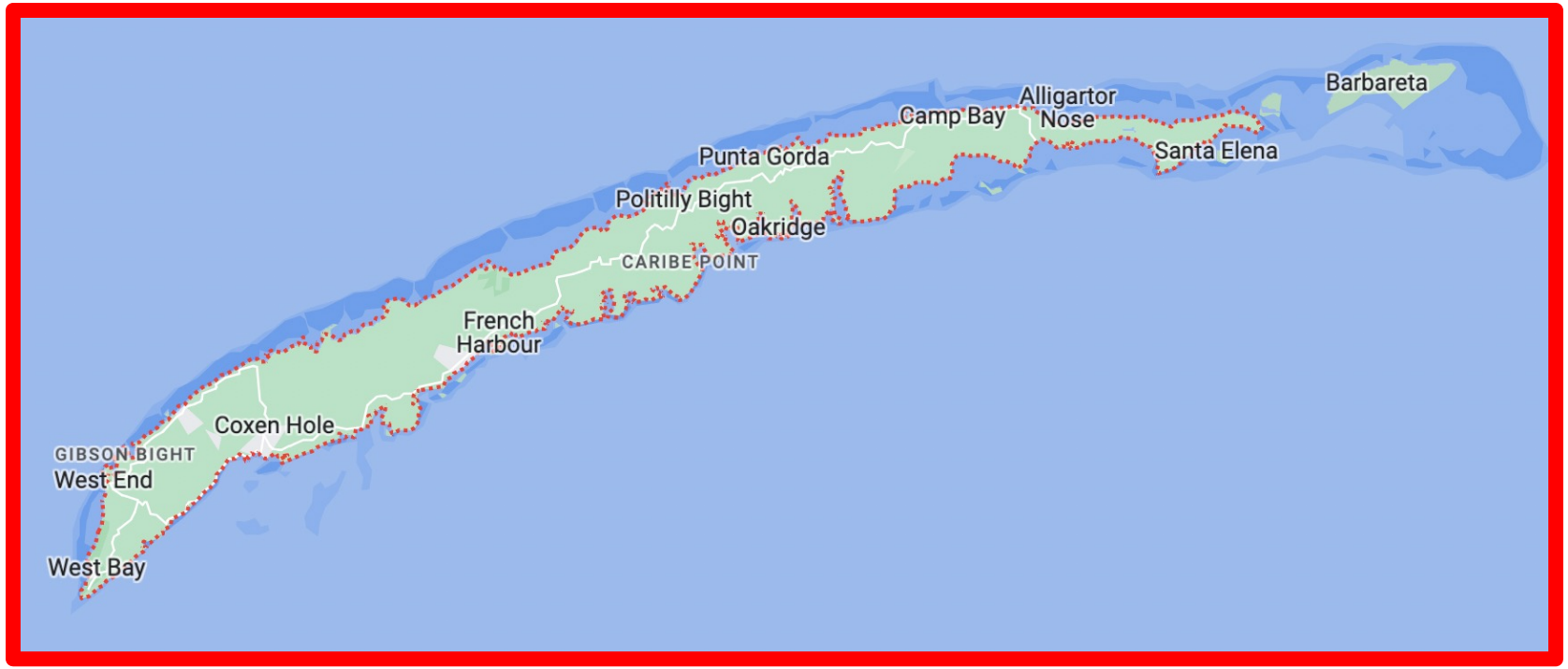
Roughly 5 orders of magnitude improvement

Test site: ROATAN, HONDURAS

~11,000 x 4,000 pixels

CORAL REEF BIODIVERSITY (IMAGE ANALYSIS)

Next Steps

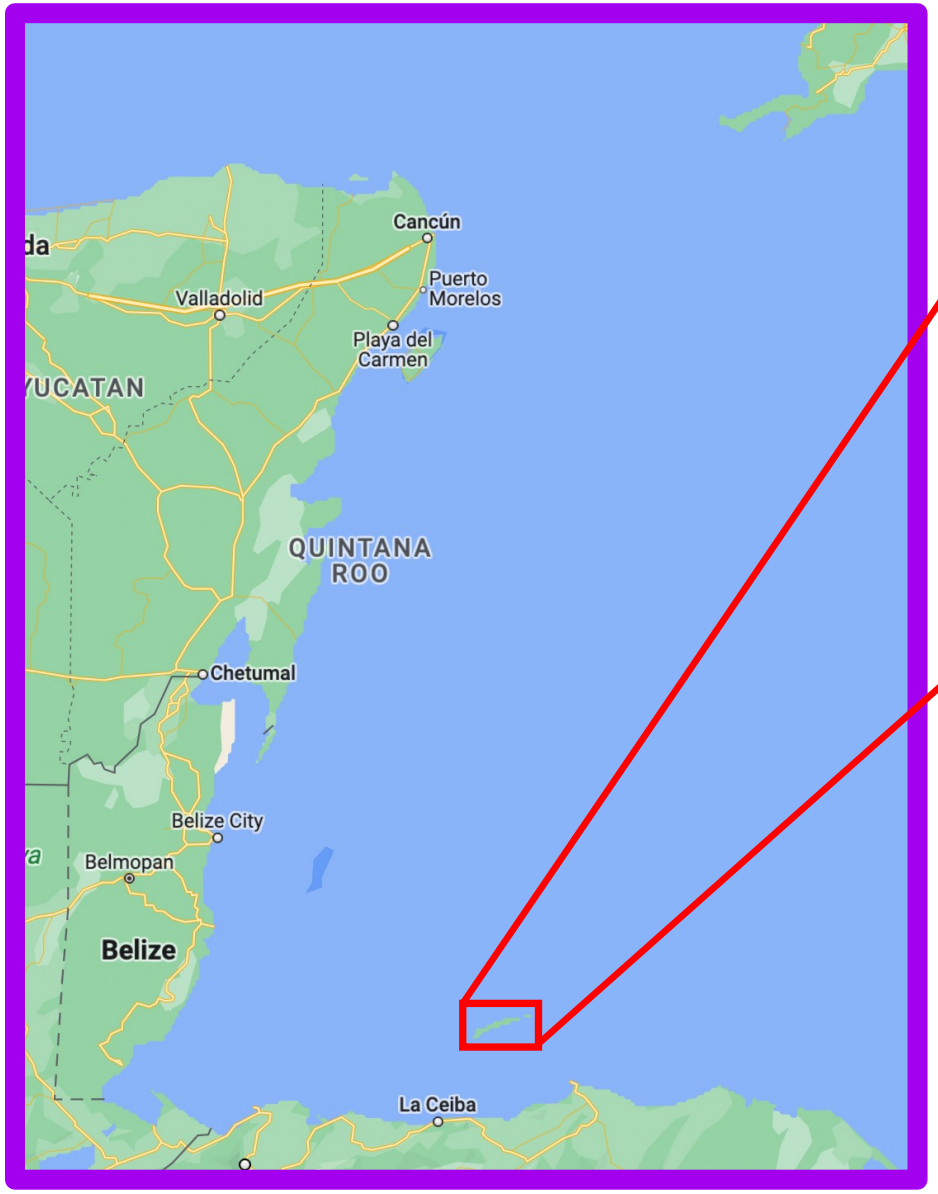


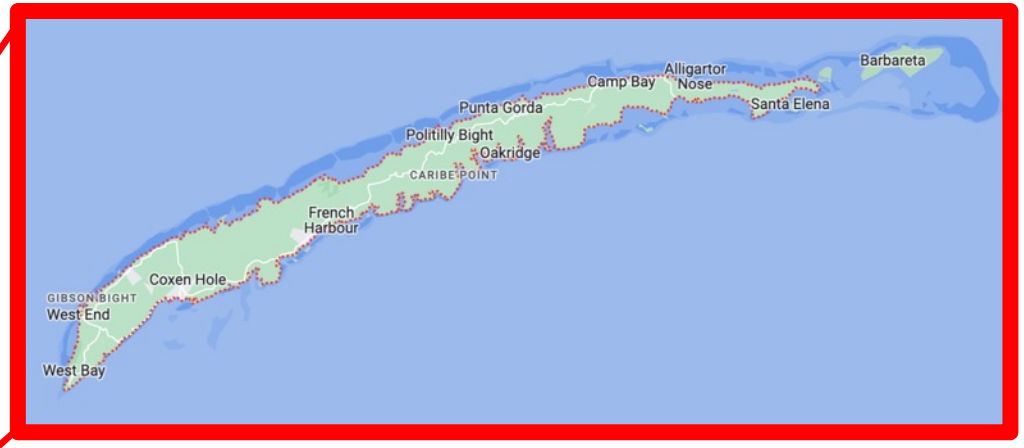
Can we try bigger regions?

CORAL REEF BIODIVERSITY (IMAGE ANALYSIS)

Next Steps

21.7 GB





49 MB

CORAL REEF BIODIVERSITY (IMAGE ANALYSIS)

Next Steps

```
coforall loc in Locales do on loc {

  const loc_d_size = d_size;
  const loc_Mask_Size = Mask_Size;

  var locD = D.localSubdomain();
  var locD_plus = locD.expand(offset);
  var locImage : [locD_plus] int(8);

  // Read in array
  var f = open(in_array, iomode.r);
  var first_point = locD_plus.first[0]*locD_plus.shape[1] + locD_plus.first[1];
  var r = f.reader(kind=ionative, region=first_point..);

  for i in locD_plus.first[0]..locD_plus.last[0] {
    for j in locD_plus.first[1]..locD_plus.last[1] {
      var tmp : int(8);
      r.readBinary(tmp);
      locImage[i,j] = tmp;
    }
  }
  r.close();
```

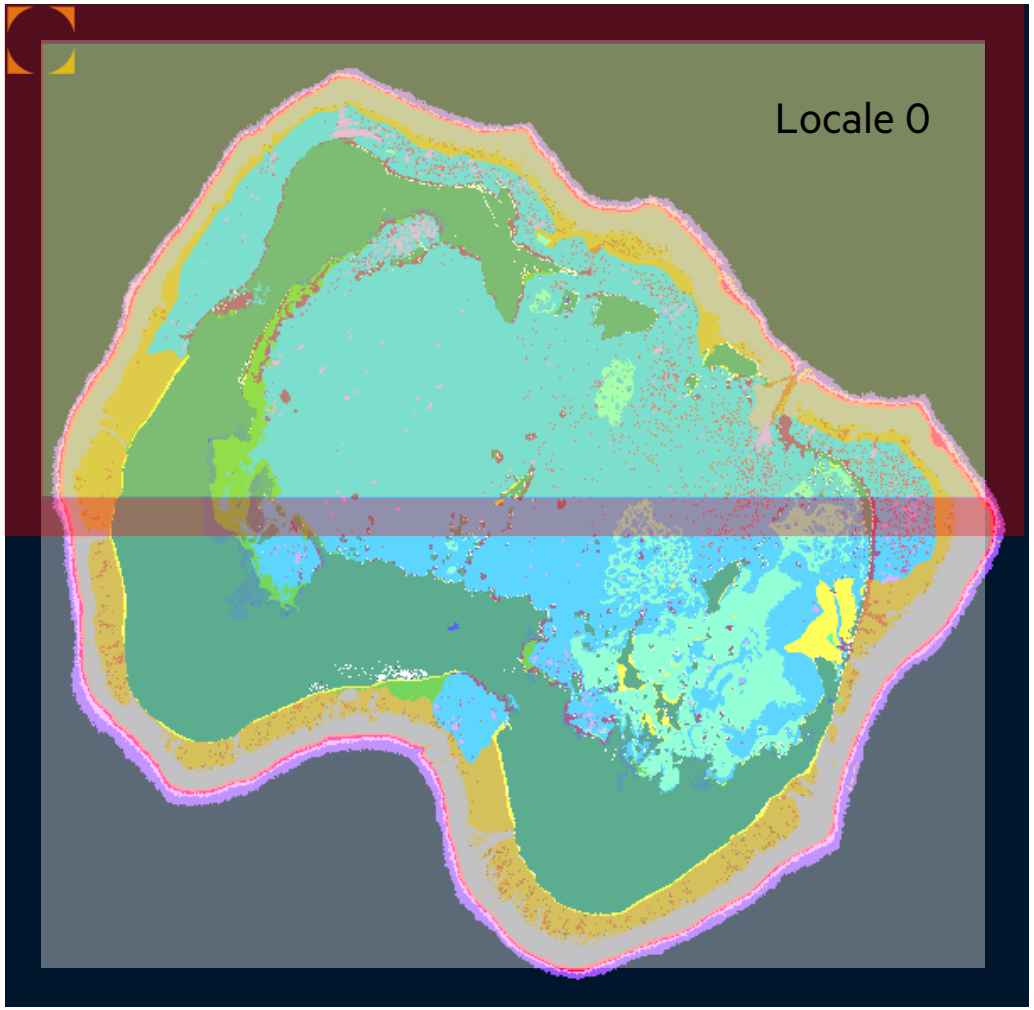
Distributed read of the input image

CORAL REEF BIODIVERSITY (IMAGE ANALYSIS)

Next Steps

```
coforall loc in Locales do on loc {

  const loc_d_size = d_size;
  const loc_Mask_Size = Mask_Size;

  var locD = D.localSubdomain();
  var locD_plus = locD.expand(offset);
  var locImage : [locD_plus] int(8);

  // Read in array
  var f = open(in_array, iomode.r);
  var first_point = locD_plus.first[0]*locD_plus.shape[1] + locD_plus.first[1];
  var r = f.reader(kind=ionative, region=first_point..);

  for i in locD_plus.first[0]..locD_plus.last[0] {
    for j in locD_plus.first[1]..locD_plus.last[1] {
      var tmp : int(8);
      r.readBinary(tmp);
      locImage[i,j] = tmp;
    }
  }
  r.close();
```

Passes the local Subdomain + a halo equal to half the mask width



CORAL REEF BIODIVERSITY (IMAGE ANALYSIS)

Next Steps

Distributed write of the output image (using C-interoperability)

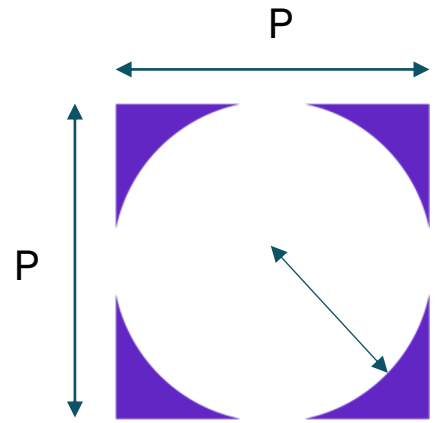
```
proc WriteOutput(filename : string, ref arr_out: [?D] real(32), ImageSpace : ?, varid
_in : int, offset : int) {


coforall loc in Locales do on loc {

  var ncid : c_int;
  var varid = varid_in : c_int;

  extern proc nc_open(path : c_string, mode : c_int, ncidp : c_ptr(c_int)) : c_int;
  nc_open( filename.c_str() , NC_WRITE, c_ptrTo(ncid));

  /* Determine where to start reading file, and how many elements to read */
  // Start specifies a hyperslab.  It expects an array of dimension sizes
    var start = tuplify(D.localSubdomain().first);
  // Count specifies a hyperslab.  It expects an array of dimension sizes
    var count = tuplify(D.localSubdomain().shape);

    var start_c : [0..#start.size] c_size_t;
    var count_c : [0..#count.size] c_size_t;

    for i in 0..<count.size {
      start_c[i] = start[i] : c_size_t;
      count_c[i] = count[i] : c_size_t;
    }

    extern proc nc_put_vara_float(ncid : c_int, varid : c_int, startp : c_ptr(c_size_
t), countp : c_ptr(c_size_t), op : c_ptr(c_float)) : c_int;
    nc_put_vara_float(ncid, varid, c_ptrTo(start_c), c_ptrTo(count_c), c_ptrTo(arr_ou
t[start]));

    nc_close(ncid);

  }
}
```

CORAL REEF BIODIVERSITY (IMAGE ANALYSIS)

Next Steps

Spectral diversity

”How much variation in the visible spectrum exists within a neighborhood?”



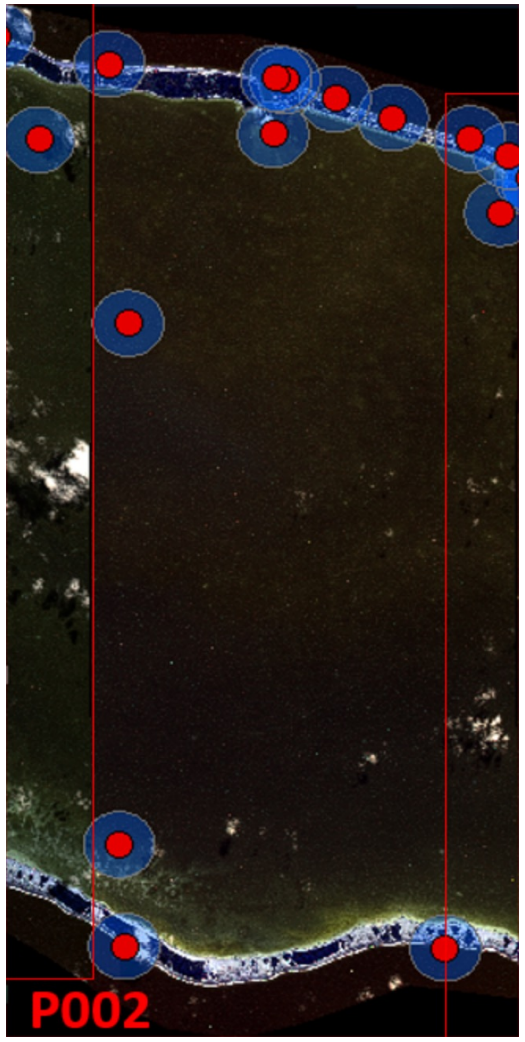
Habitat diversity: Count the number of points of each type

Spectral diversity: For each point, compare spectrum *against every other point*

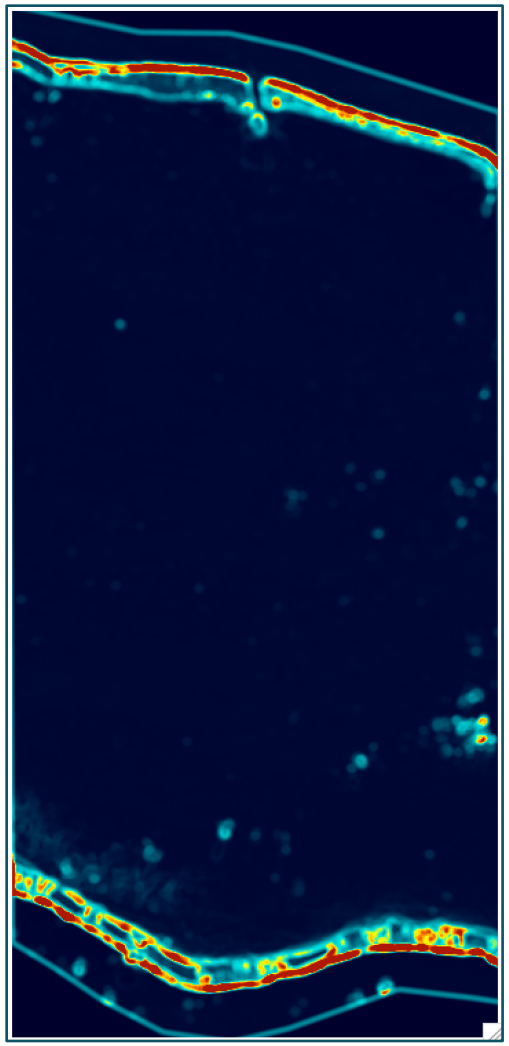
Habitat diversity: $O(MNP)$

Spectral diversity: $O(MNP^3)$

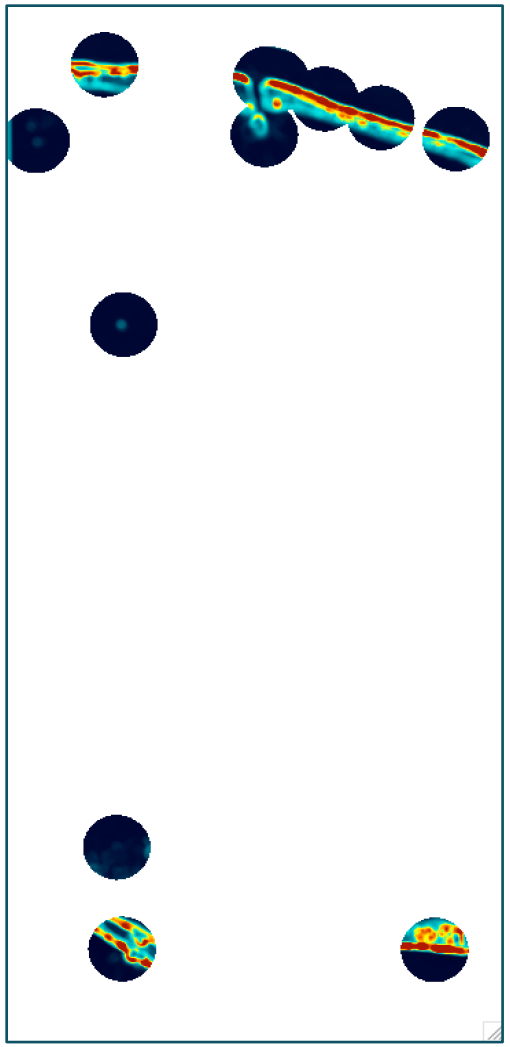
Spectral diversity / heterogeneity



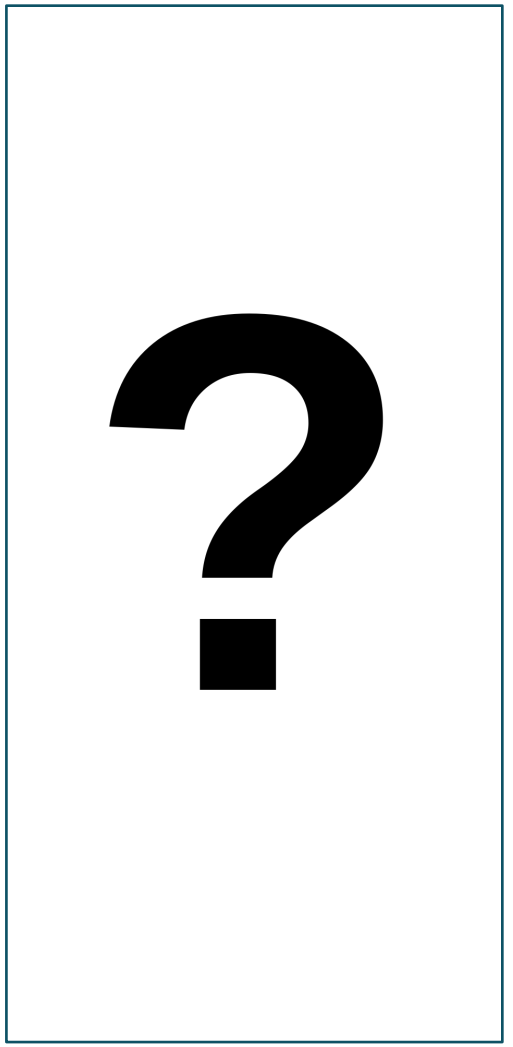
Rangiroa, French Polynesia
19,639 x 9,015 (8 color bands)



Full map
~ 12 hours on 432 cores



Dive sites only
~ 1.5 hours on 432 cores



GPU version
?

CORAL REEF BIODIVERSITY (IMAGE ANALYSIS)

Next Steps

Spectral diversity ***on GPU***

```
foreach i in centerPoints.dim(0) {
//for i in centerPoints.dim(0) {
  assertOnGpu();

  var tmpLL : real = 0;
  var tmpLC : real = 0;
  var tmpLR : real = 0;
  var tmpCC : real = 0;
  var tmpCR : real = 0;
  var tmpRR : real = 0;

  calc_distance(Array, Masker.left(), Masker.left(), tmpLL, bs, be, i, first_point);
  calc_distance(Array, Masker.left(), Masker.center(), tmpLC, bs, be, i, first_point);
  calc_distance(Array, Masker.left(), Masker.right(), tmpLR, bs, be, i, first_point);
  calc_distance(Array, Masker.center(), Masker.center(), tmpCC, bs, be, i, first_point);
  calc_distance(Array, Masker.center(), Masker.right(), tmpCR, bs, be, i, first_point);
  calc_distance(Array, Masker.right(), Masker.right(), tmpRR, bs, be, i, first_point);
```

```
Distance circle has a radius of 5 points.
Elapsed time at start of coforall loop: 6e-05 seconds.
Starting coforall loop.
Distance circle has a radius of 5 points.
Made masker on 0
Starting convolution at 18.5078.
Before gpu 6..15236
6
10066
0 (gpu 0): main_GPU_iter2.chpl:38: kernel launch (block size: 512x1x1)
Took 0.005122 seconds to complete convolve.
Elapsed time to finish coforall loop: 18.6655 seconds.
```

CORAL REEF BIODIVERSITY (IMAGE ANALYSIS)

Summary

- Chapel program to estimate reef biodiversity using satellite imagery
  - Embarrassingly parallel and scalable
  - Multiple orders faster than incumbent MATLAB program
  - *Very* exciting for my colleagues in the conservation world!
- Takeaway? Lots of applications out there (not necessarily complicated) that could benefit from what Chapel provides