How to Replicate FB’s distribution grid prediction model

Brandon Rohrer

Sedar Sahin

Dimitry Gershenson

Anna Lerner

Grid modeling problem

- Using open and public accessible data sets, estimate the location of the medium voltage power lines (distribution grid).
- The poles carrying medium voltage power lines are too small to be automatically identified in satellite imagery, and substations are too varied in appearance.
- As a result, we rely on indirect indicators of electrification and determine likely paths for the power lines that connect them.

Data sets used

- VIIRS day-night band “night lights”
  - NASA-provided monthly composites covering 2012-2106
  - 5 years * 12 months = 60 in all
- The MODIS land cover data set
- Open Street Map highways

Nightlights

- The most reliable publicly available indicator of electrification is nighttime radiance, collected by the VIIRS day-night band sensor on the SUOMI-NPP satellite.
- Using these “night lights”, we can see which communities regularly generate enough light to be detected from space.
  - Images are collected about 2am local time, which means they are dominated by all-night outdoor lighting.

Assumptions

- By using night lights to infer electrification, we make a two assumptions:
    1. That all settlements on the grid generate detectable lighting,
    2. That all settlements off the grid (including those on micro-grids) do not.
- We are continuing to challenge these assumptions against the electrification data we can find, but so far they have not proven to be grossly inaccurate.

Nightlights processing

- Using a 311 * 311 km area
- For each monthly image, if more than 20% of the pixel values were missing (due to cloud cover) reject the image.
  - Missing values are indicated by pixels values exactly equal to zero.
- Interpolate the data to a regular grid.
  - We used zoom level 17 Bing tiles.
    - Each grid cell “pixel” is approx. 304 m * 304 m.
  - Flood fill gaps with nearest neighbor values.
  - In case of collisions, keep the maximum value.

Nightlights processing

- Human-generated lighting is just one source of night light radiance. In order to make use of the images, they must first be corrected for reflected starlight and moonlight, clouds, wildfires and lightning.
- Indirect and non-settlement sources of light need to be removed as well, including oil well flares fishing boats, bonfires and reflected city lights.

Nightlights processing

- We work with a sequence of monthly composite images, covering a period of five years.
  - NASA composites an image each month, combining the most cloud-free, moonless images patches to cover the earth.
- We apply a custom image processing filter to each image that subtracts the background level out and identifies pixels that are significantly brighter than those in their immediate neighborhood.
  - This eliminates background and reflected light.
- Then, we find which pixels are consistently lit over time.
  - This eliminates wildfires and temporary or mobile light sources.
- The points that remain are assumed to be electrified settlements.

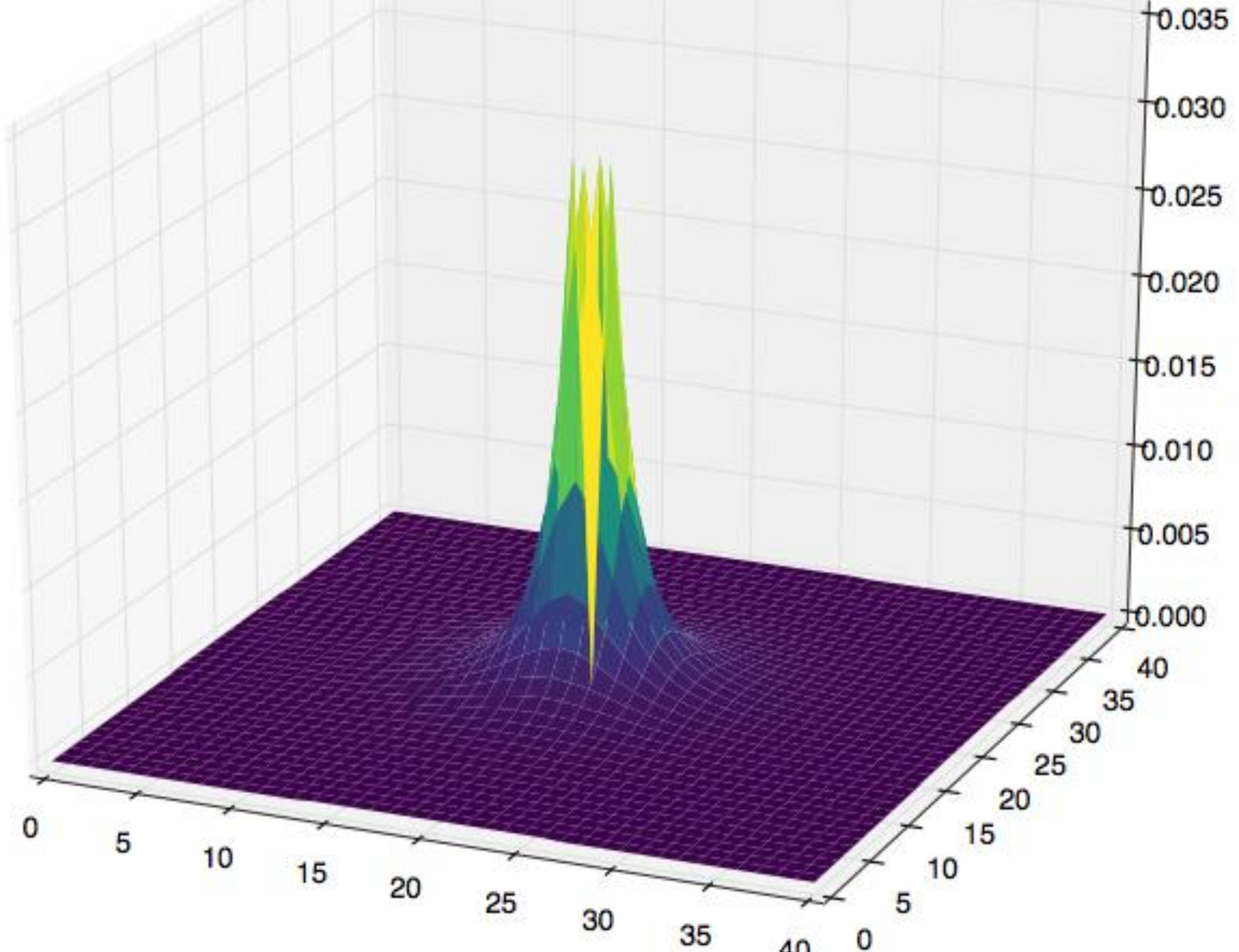
Nightlights filtering method

1. Estimate background lighting levels by convolving a 2-dimensional filter with the image.
2. Filtered pixel values = original pixel values – background – threshold
   - A threshold of .1 was selected through trial and error
3. Keep only the positive pixel values. Set the rest to zero.

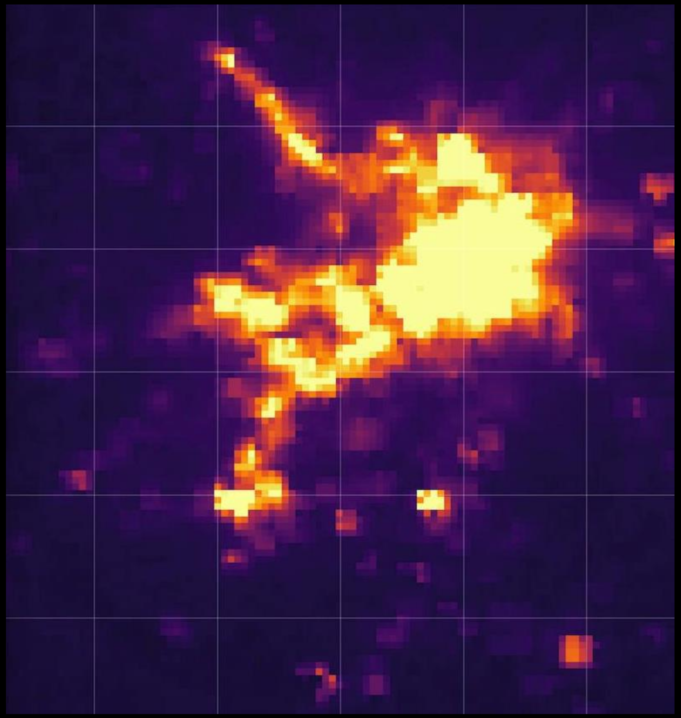
Filter design

- Filter is 41 by 41 pixels, centered and symmetric.
  - Each pixel is about 304 x 304 meters.
  - The filter covers about 12.5 x 12.5 km.
- The value for each pixel is 1 / (1 + d/2)^3, for d <= 20
- where d = sqrt(d_rows^2 + d_cols^2)
- where d_rows is the number of rows to the center pixel and
- d_cols is the number of columns to the center pixel.
- The value of all other pixels is zero.
- The value of the center pixel is also set to zero.
- The filter is then scaled so that the sum of all filter pixel values is one.

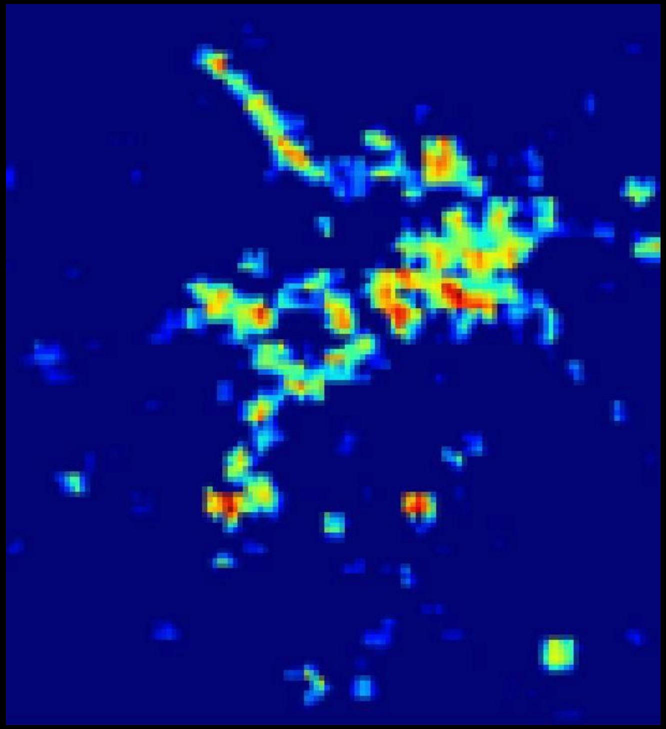
Filter visualized



Single image, before and after filtering



Before
Includes lights, reflected illumination, and background variations.



After
Dark blue shows areas of no night lights.

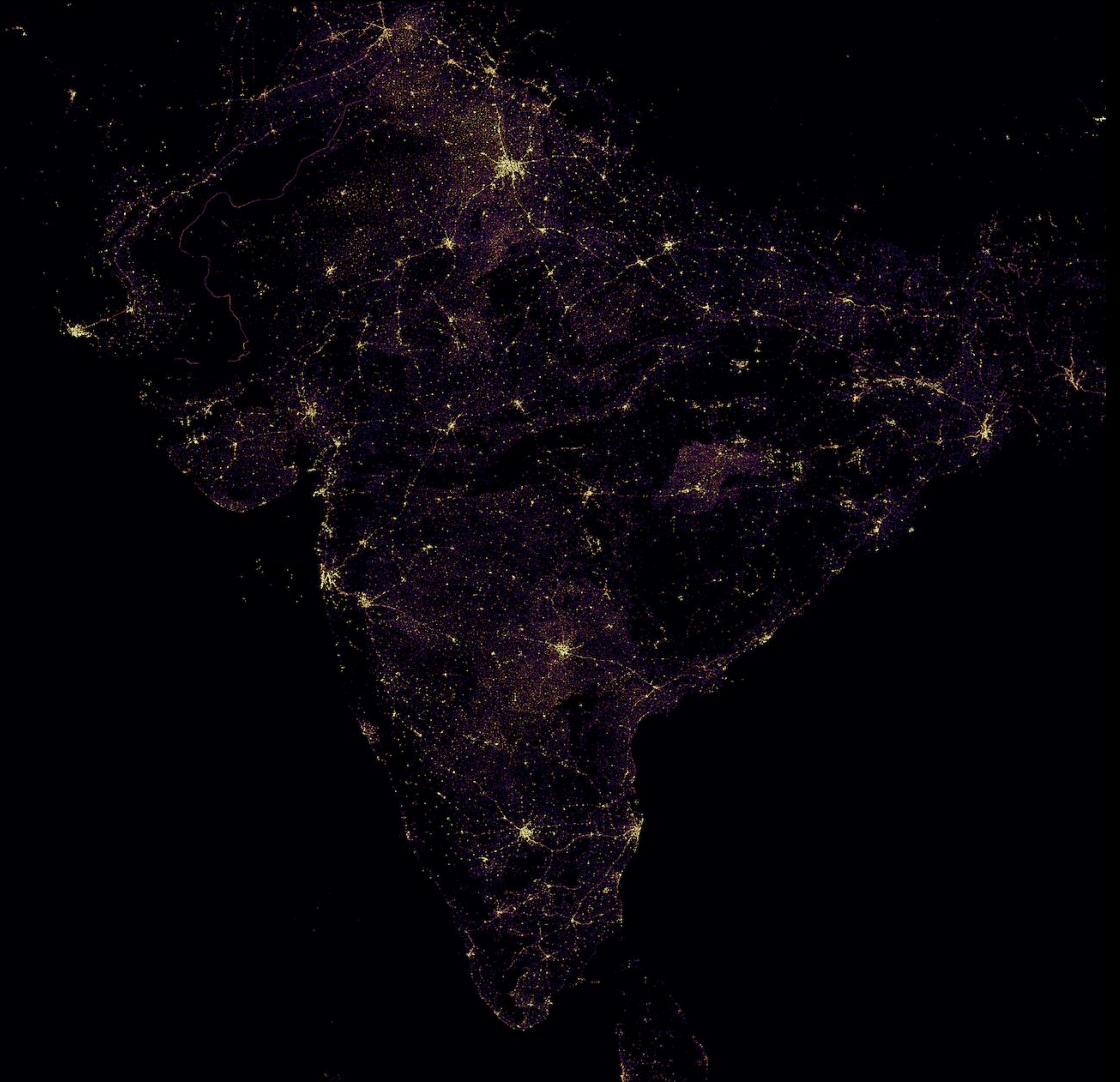
Find persistently lit tiles

- For each zoom level 17 tile, across all 60 filtered monthly composites, find the 70th percentile value.
- This means than a tile must appear lit at least 30% of the time to register as electrified.
- This helps eliminate transient lights and illumination due to causes other than electrically-powered lighting.

Account for land cover type

- The MODIS land cover data set indicates, among other things, watery areas where settlements are unlikely.
- Original resolution is 500 m by 500 m.
- Interpolate to Bing zoom level 17, filling empty tiles with nearest available data.
- Remove persistent lights from tiles containing water and permanent wetlands. These are most likely reflections or deep water oil rigs.

Sample results



Grid path selection

- Using the electrified settlements, the grid estimation algorithm seeks to connect them in the most efficient way possible. Using known electrical grids as templates, grid paths are encouraged to follow roads and prefer shorter paths.
- The process resembles a dot-to-dot.

Path cost

- For path planning, break the region up into a fine grid and assign a "cost of visiting" each grid cell.
- All else being equal, paths visiting fewer cells will have lower cost.
    - Shorter paths will be preferred.
- Cost of visiting is low for roads.
    - Low cost of visiting means the grid will tend to follow roads, a strong trend in the mapped grids we've investigated.
- The larger the road, the lower the cost of visiting.
    - Everything else being equal, the grid will follow larger roads rather than smaller ones.

Open Street Maps highways types

- OSM has a few different types of highways, classified by purpose and amount of use. Lower weights (cost of visiting) are assigned to larger ones.

Highway type	weight
Motorway	1 / 10
Trunk	1 / 9
Primary	1 / 8
Secondary	1 / 7
Tertiary	1 / 6
Unclassified	1/ 5
Residential	1 / 4
Service	1 / 3
No highway	1

Discretize roads

- OSM roads are stored as vectors. Rasterize them, that is, reduce them to a regular tiling.
- We use Bing zoom level 20 tiles (about 38 m by 38 m).
- OSM roads are traced and marked every 3.33 meters.
- Tiles containing markers are considered part of that road and assigned the associated weight.
- Tiles containing markers from more than one road keep the minimum weight.

Create targets

- All the electrified Bing zoom level 17 tiles also need to be re-expressed as zoom level 20 tiles.
    - Each level 17 tile contains 2^3 * 2^3 or 64 level 20 tiles.
- These level 20 tiles showing persistent night lights are the “targets”, the locations assumed to be electrified that the grid must connect.

Seed the origins

- “Origins” are the set of all locations believed to be on the grid.
- Initially there is no grid, so there are no origins.
- Randomly select an electrified location to serve as the initial origin.

Pathfinder

- Pathfinder is a modified version of Dijkstra's algorithm for finding the shortest distance between two points in a graph. It differs in a few important ways
  - It finds the shortest distance between a target point and the nearest of a set of origin points. This is then repeated for each target point.
  - It assumes a gridded topology. In other words, it knows that each node only touches its neighbors to the north, south, east, west, northeast, northwest, southeast, and southwest.
- Like Dijkstra's Algorithm, Pathfinder assumes that all weights are 0 or greater.

The halo

- The halo is the set of points under evaluation. They surround the origin points and expand outward, forming a growing halo around the set of origins that eventually envelops targets.
- It is implemented using a heap queue, sorted by distance, so that the halo point nearest to an origin is always the next one that gets evaluated.

Distance

- The distance between the grid so far and any other tile is the sum of the visiting costs on the shortest (least expensive) path between the grid so far and that tile.
- Paths can traverse north, south, east, and west. They can also traverse northwest, northeast, southwest, and southeast, but when doing so, they incur a visiting cost that is weighted by sqrt(2), to account for diagonally traversing a square grid cell.

The algorithm

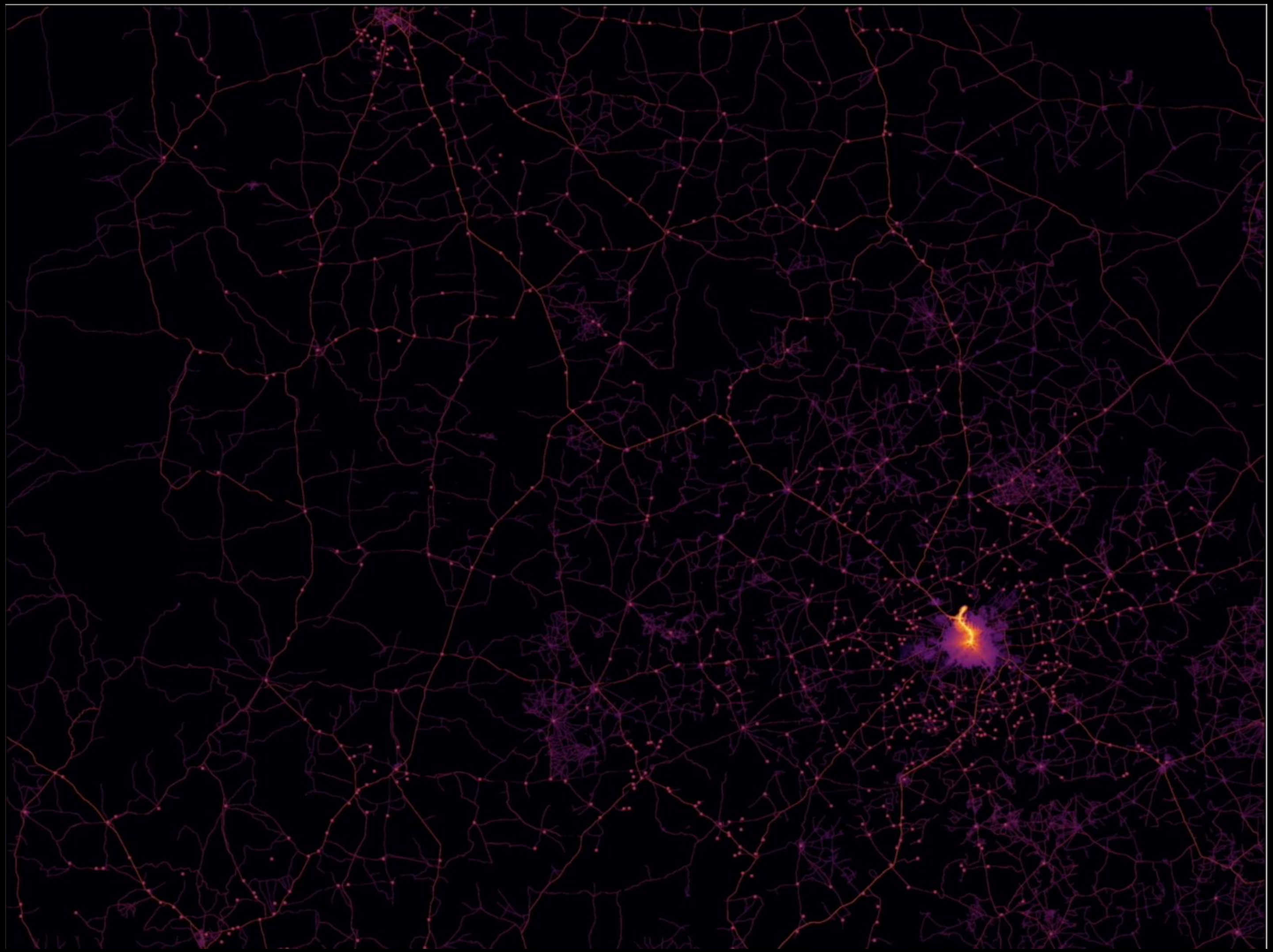
- Set the distance of the origin to zero and push it to the halo.
- Repeat
  - Pop the lowest distance point in the halo.
  - Check all its neighbors (N, S, E, W, NW, NE, SW, SE)
    - If distance to the neighbor through the current location is lower than the previous shortest distance to the neighbor, update it.
  - Push any first-time visited locations to the heap.
- Until no targets remain to connect

The algorithm, part 2

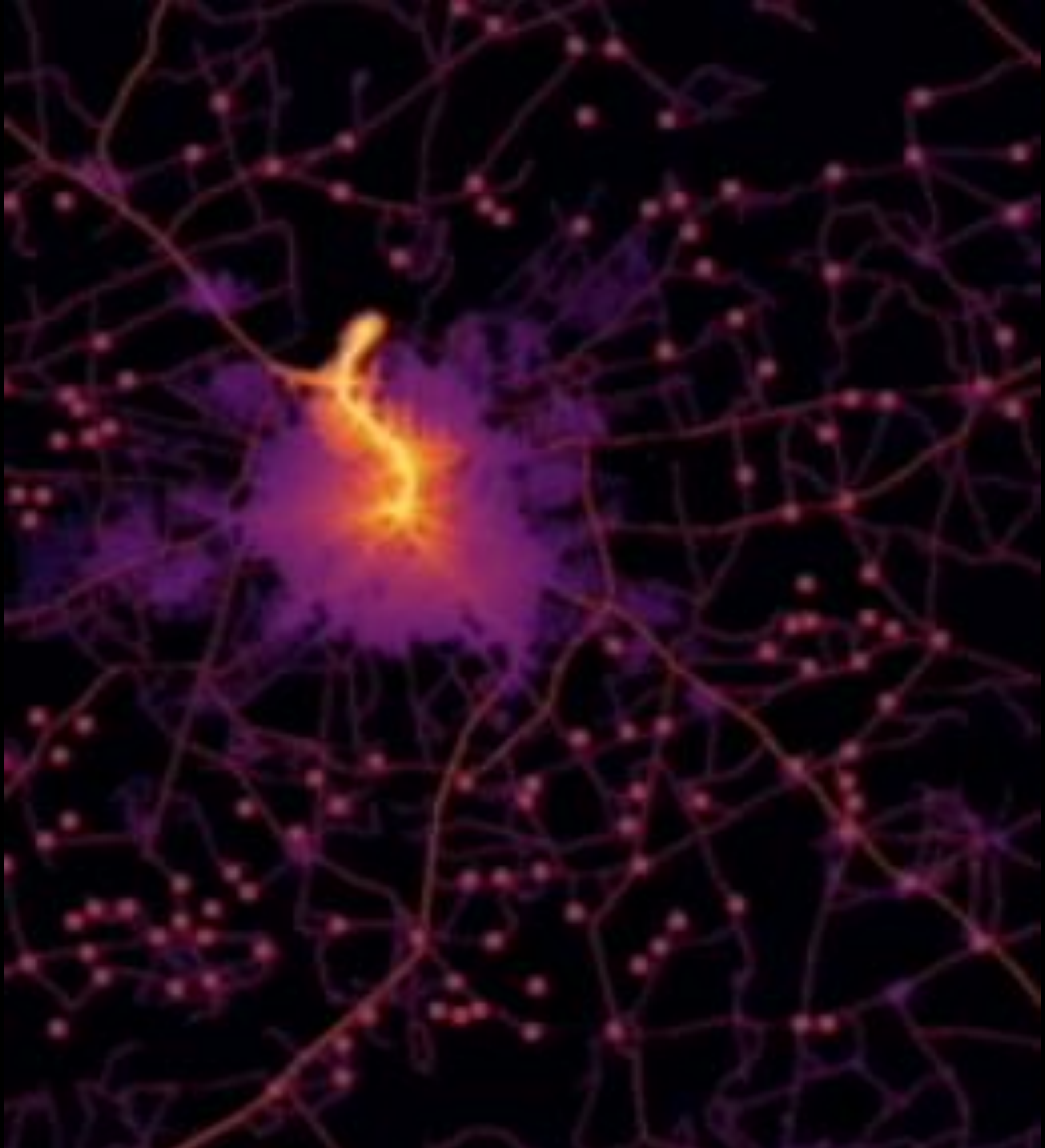
- Whenever a target is discovered
  - Set its distance to zero.
  - Push it to the heap.
  - Do the same with all the points in the shortest path connecting it to the existing grid.

Pathfinder algorithm in action

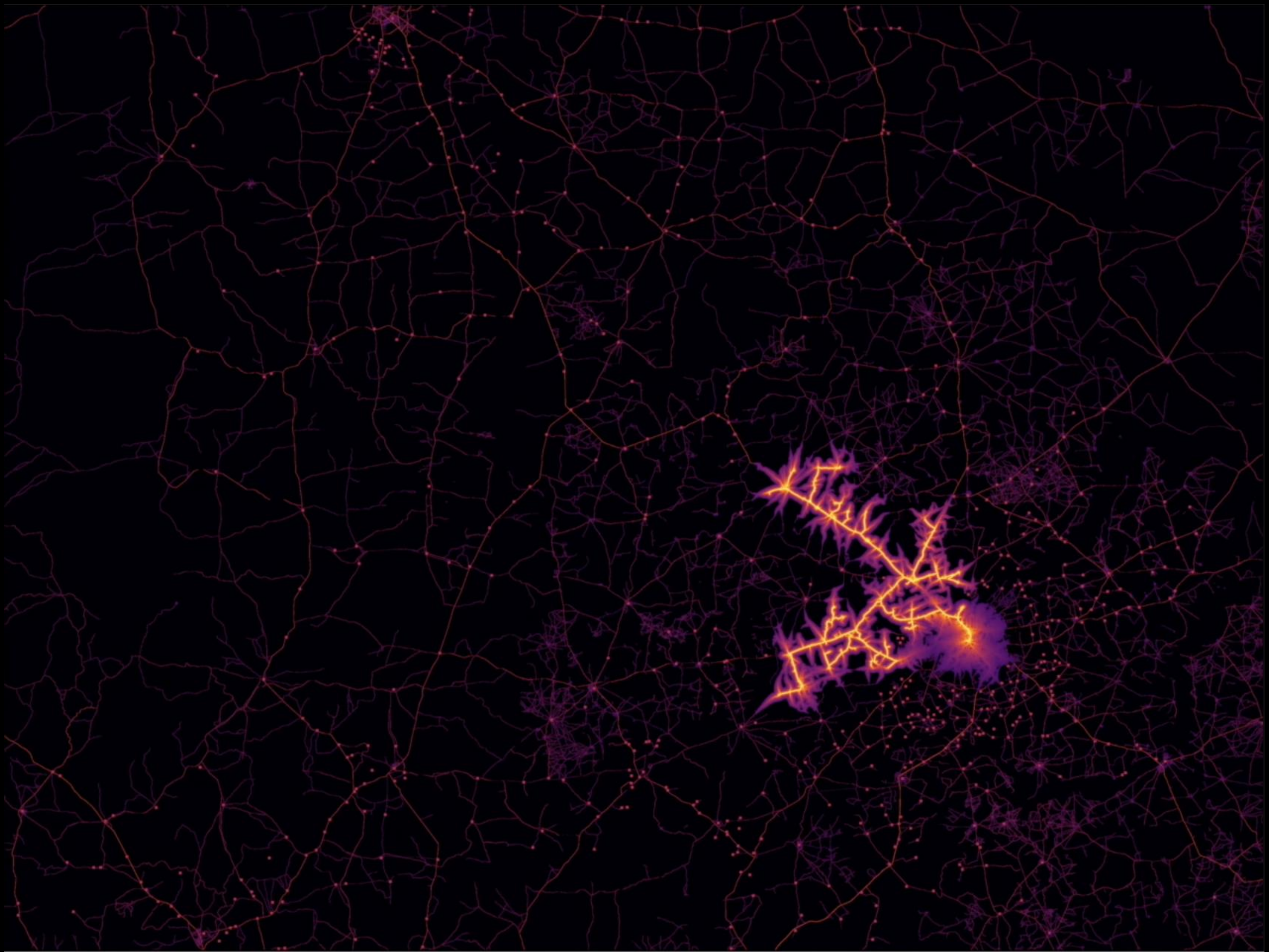
- Red squares are targets (electrified settlements)
- Roads are overlaid.
- Bright yellow shows grid so far.
- Pink shadow around it shows the extent of the halo.



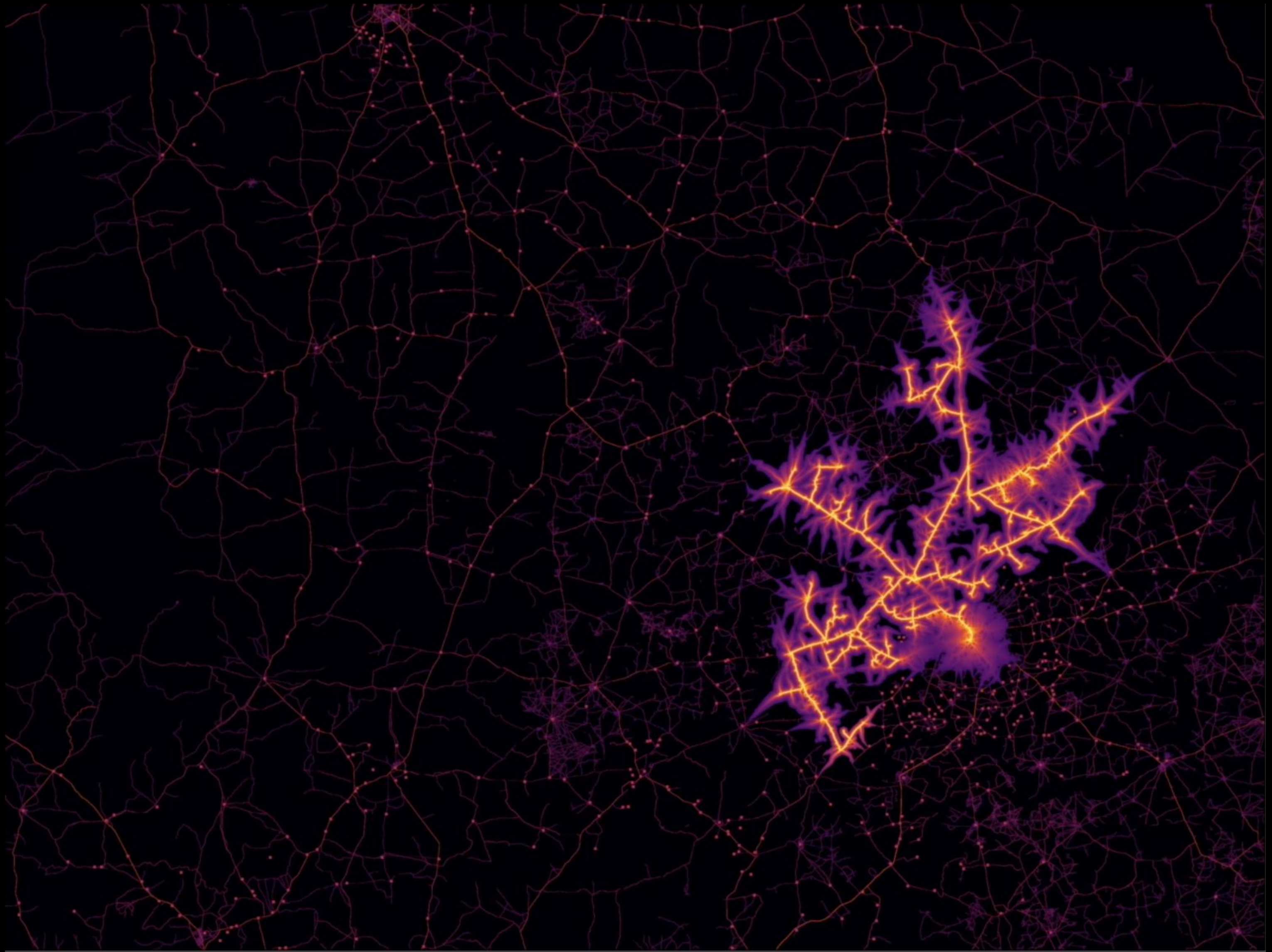
Closer look



Pathfinder algorithm: expansion

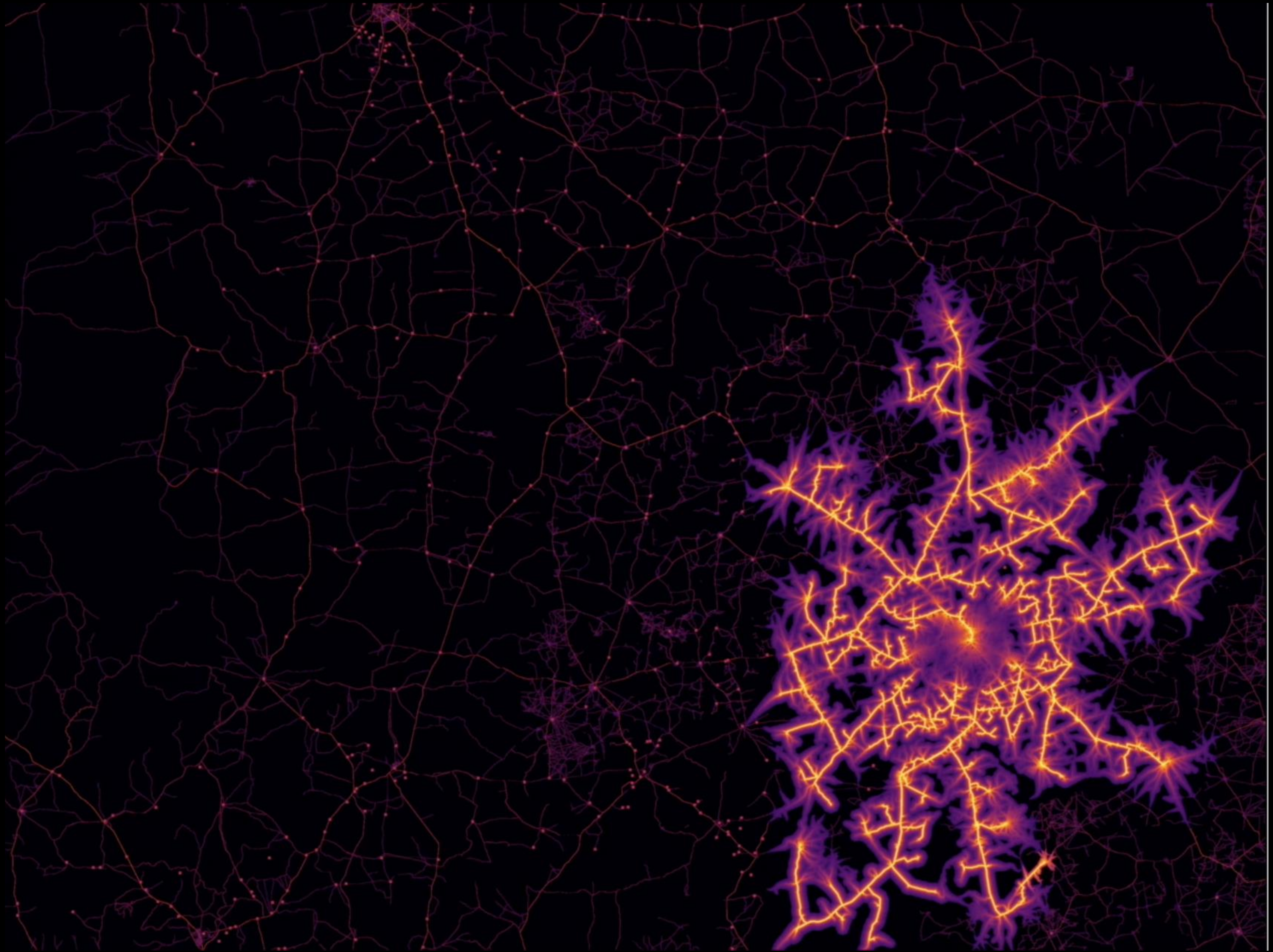


Pathfinder algorithm: expansion



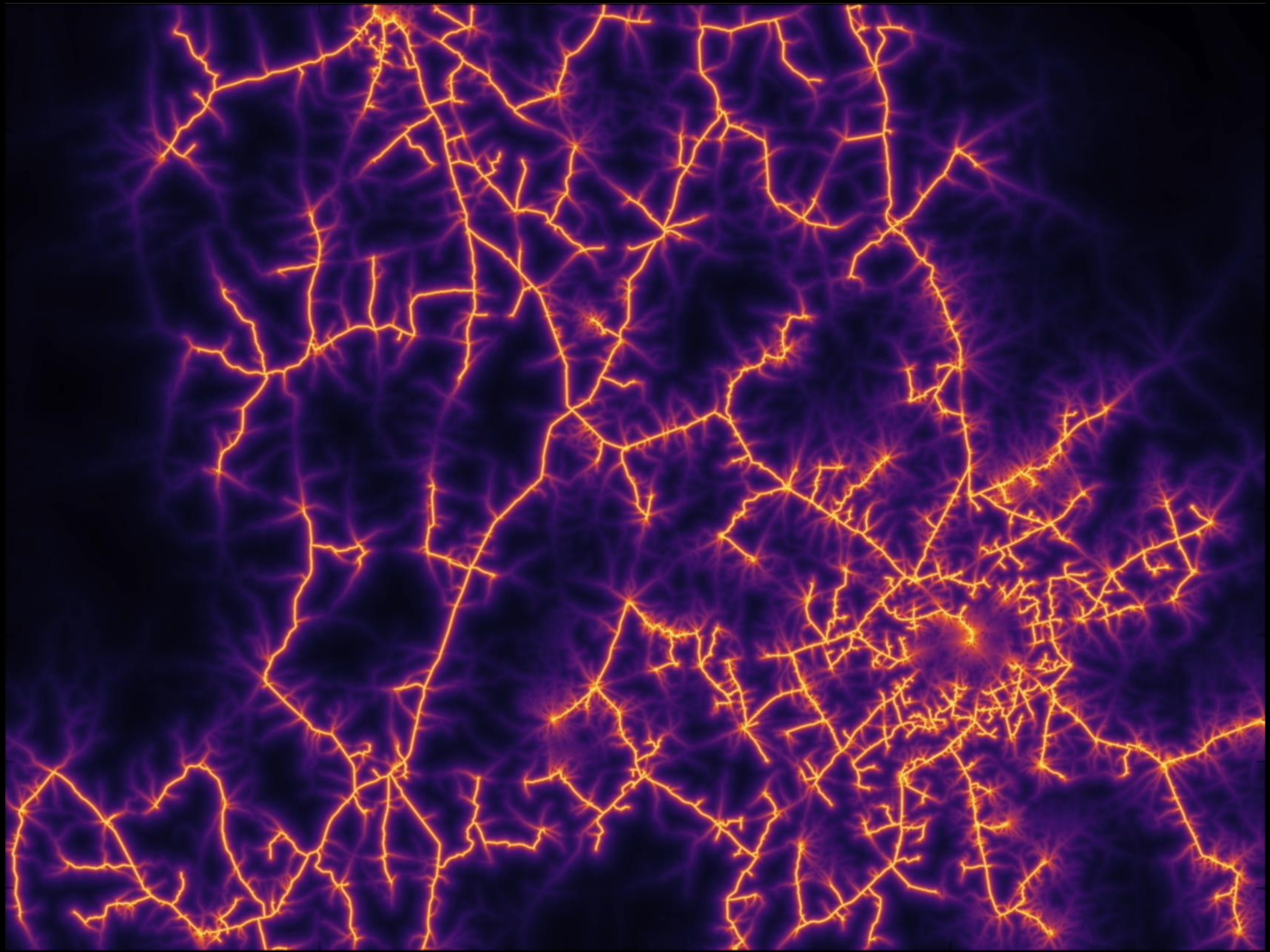
Pathfinder algorithm: expansion

- The halo quickly expands, strongly preferring major roadways.
- The grid follows.



Pathfinder algorithm: completion

- By the time the last electrified settlement has been reached, most of the region has been encompassed by the halo.



Form of results

- When complete, the grid estimate is represented as a set of contiguous Bing zoom level 20 tiles.
- Compared to ground truth in Nigeria and Uganda, the estimate is within 1 km of the actual grid 65% of the time.

In some places, the model performs quite well

- Blue is ground truth.
- Bright yellow is modeled grid.
- Through most of this region, blue lies exactly over yellow.