List of Potential Ideas:

* Analyzing celtic knots
  + Analyze road junctions from images
    - Potentially generate junctions based off points
    - Find points of intersection
  + Techniques
    - Identifying parallel lines - match by gradients
  + Existing research
    - <https://www.semanticscholar.org/paper/Computer-Generated-Celtic-Design-Kaplan-Cohen/66f2c89488c7bb7530aa0cccfeff529663e4505f>
  + Potentially helpful research
    - <https://www.pnas.org/content/104/42/16432>
    - <http://robertmatthews.org/wp-content/uploads/2016/03/RAJM_SSR_tangling_2015.pdf>
    - <https://users.cs.duke.edu/~pankaj/publications/papers/writhe.pdf>
    - <https://www.sciencedirect.com/science/article/pii/S0379073820301341>
* What would happen if you tightened a knot vs loosened a knot?
* Identifying animals in an image
  + <https://www.pnas.org/content/115/25/E5716>
* Identifying deepfakes
  + <https://adversarialdeepfakes.github.io/>
* Genome sequence analysis
  + Edit distance (2d space algo)
  + Classify sequences and then apply existing string algorithms
* Problem statement: Clearly state the goal of your project. When someone uses your system, what is the expected input to the system, and what is the desired output?
  + Identify the splines and intersections in celtic knots
  + Why is this interesting?
    - Have to understand the intersections and idea that paths are crossing (occlusion)
  + Stretch goal: artificially tighten/loosen the knot
* Approach: Describe the technical approach you plan to employ.
* Experiments and results: Describe the experimental setup you will follow, which datasets you will use, which existing code you will exploit, what you will implement yourself, and what you would define as a success for the project. If you plan on collecting your own data, describe what data collection protocol you will follow. Provide a list of experiments you will perform. Describe what you expect the experiments to reveal, or what is uncertain about the potential outcomes.

Research papers

* Celtic knot generators
  + Many of them are tile based.
  + <http://hypatiastudio.com/celticknots/>
* Detecting parallel lines: people on the internet recommend using Hough transform to get the lines. Then use line properties to find parallel lines
  + Maybe include some proximity value; prioritize pairing (an assumption) to get the right pairs of lines.
  + Find the line going through the center
* Detecting parallel curves:
  + <https://www.worldscientific.com/doi/abs/10.1142/S0218001496000475>
    - Couldn’t find how to download the paper
  + <https://www.sciencedirect.com/science/article/pii/S107731429790552X?via%3Dihub>
* Detecting intersections
  + Claudia:
    - One heuristic is to just first use Hough Transform to find all the lines
    - Then use those lines to get the “intersections” based on parallel and proximity
    - Each true intersection has 4 points
    - If the widths are constant, just take the middle of the square
    - Derive some points of curvature? Extrapolate the splines
    - <https://docs.scipy.org/doc/scipy/reference/interpolate.html>
* <https://www.researchgate.net/publication/37923699_Picking_knots_from_trees_-_The_syntactic_structure_of_Celtic_knotwork>