**Assumptions & Implications**

* “Collection protocol has produced images all of one flank of the animals, avoiding the ambiguity associated with seeing two different sides of the same animal.”
  + This poses a challenge. We hope to combat this by associating images of different sides of a given animal such that we can identify an animal from any angle
* “Fully addressing the issues of a [sic] efficiently searchable, large-scale, and dynamic database of animal images is beyond the scope of this paper”
  + This may not scale well to Panthera’s proposed database. Possible future work: interface with Panthera’s proposed database similar to IBEIS
* “We are only interested in correctly identifying the animal (high precision) not finding all matching images (total recall)”
  + This allows the algorithm to work faster than one intended to find ALL matches, but may be disadvantageous for training animal-image associations

**Algorithm**

Note: The following section outlines the One-vs-One matching, which is not implemented. HotSpotter uses a One-vs-Many matching algo, which will be covered later.

Let:

|  |  |
| --- | --- |
| Name | Representation |
| I | Image, either from the database or a query |
| x | Locations of spatial and scale extrema of the Hessian-Hessian operator applied to I (features) |
| A | Matrix that represents ellipse which fits hot spots at locations x. A = [a,0;b,c]. May represent a transformation from points on the ellipse to points on the unit circle |
| 𝜃 | Orientation angle of ellipse defined by A. “Often the aggregate of gradient direction in the region surrounding x, but instead we assume the the “gravity vector” is downward in each image and use this to simply assign 𝜃=0” |
| *X* | {x\_i} |
| *A* | {A\_i} |
| *d* | 128-dimensional descriptor for each region (x,A). Calculated with RootSIFT (componentwise square root of the SIFT vector) |
| *D* | {d\_i} |
| I\_D & I\_Q | Database image and query image |
| *D\_D & D\_Q* | Set of RootSIFT descriptor vectors for database image and query image |
| *q\_j & q\_j2* | Two query image descriptor vectors closest to found (with high probability through a restricted, prioritized search) |
| *r\_{i,j}* | Ratio, given by |
| M\_D | Set of matches of form {(i,j,r\_{i,j})} |

Given all these, the ith feature of I\_D and jth feature of I\_Q are associated if the ratio r\_{i,j} exceeds some threshold t=1.6^2. Such matches are collected into M\_D. Similarity between images is calculated with a simple sum, encouraging MANY DISTINCT matches

Utilizes spatial re-ranking, which doesn’t make a huge difference, apparently.