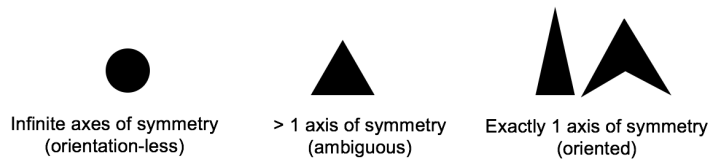Detailed info on the features below:

- *Creating "agents"* or predefined 2D shapes of arbitrary size and color, where users can tether the "face" direction of the agent to the direction of its movement, or allow its orientation to vary either randomly or programmatically with respect to its direction of movement. Agents can appear to be animate (e.g. people or animals) or objects (e.g. a ball) as seen from the top). Some examples below:

| Infinite axes of symmetry (orientation-less) | > 1 axis of symmetry (ambiguous) | Exactly 1 axis of symmetry (oriented) |

- *Assigning "motivators"* or motion algorithms for the aforementioned shapes within a frame. The trajectory can be specified programmatically, by taking in user-provided constraints on (speed, direction, acceleration, distance from edges and/or other shapes, etc.) or be set manually by entering coordinates. The program incorporates physics libraries so that the animations appear "realistic" in terms of Newtonian physics as well as biological motion (as much as possible).
  - With these motivators, we have simulated two-person interactions such as predator-prey, playful vs, aggressive interactions, agents arriving at a goal etc. in the animation mode, and with the options to allow users to control one of the agents when the other one is interacting with them [gaming mode].
  - Example motion parameters that we vary within an agent include:
    - speed, direction, acceleration, noisiness (jitter) – these can be manipulated parametrically
    - Agents' movements can also be tethered to others flexibly – for example, with varying degrees of noise (example "chasing_subtlety.png"):
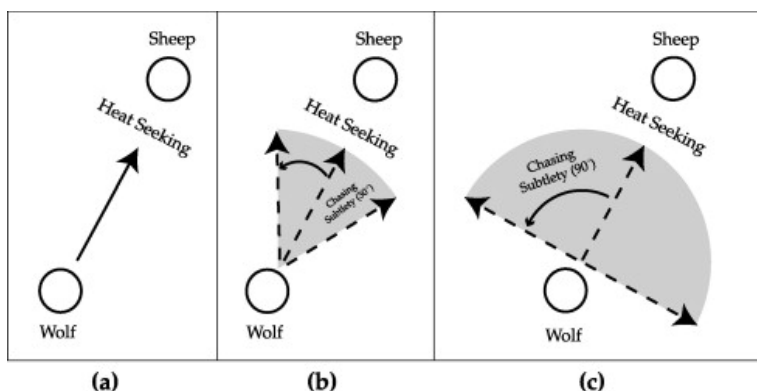
- Check out how the app GUI works here, example videos here
- In the gaming mode, the user can control one of the agents with the keyboard or the mouse (example here, use W-A-S-D to move the agent).