1. Information loss and neural computation generally go hand in hand. When cells are specialized to select a certain feature everything else included is lost except that feature such as direction sensitive cells. These cells will only activate based on motion direction everything else is irrelevant-task specificity. (150-151). Mechanisms are in place to reduce noise. Noise isn’t relevant information the cells need to spend time computing which is why most cells are action potential based that require a certain threshold to be met before activating to send information. (151) A third example of lost information during computation is motion extrapolation. In computing anticipation, if there is an abrupt change in direction the ganglion cells do not react at the same moment.(155) it is delayed in representing correct path, if there are many consecutive abrupt changes to movement I would think some information would be lost and would correspond to the last changes before the path smooth back out. (I might be wrong with this assumption)

2. One method to reduce noise is through temporal filtering which enhances light signal frequencies while suppressing others resulting in more signal and less noise to affect the signal. (152) Noise is also present in motion of head, body and eyes which cause the image perceived to be jittery but this is reduced due to a vestibular - optic reflex pathway that is dedicated to processing just this motion. Which does reduce the noise but does not completely rid of it.(152) The third method, rod bipolar cells reduce noise by averaging out the noise it pools from rod photoreceptors. Somewhat like a threshold mechanism, cells with higher activation will stand out from the ones with smaller activations. Pooling allows to interpret various activation levels from a group of cells rather than a summation of a group of cells. (153)

3. Rectification is a thresholding mechanism. Once a cell reaches a certain threshold generally presynaptically then the cell will fire. It allows for the activated cells to still communicate its activation after a certain threshold is reached, like an OR gate in that if there are inhibitory cells the overall response won’t be inhibitory. This is exhibited in the paper with moving texture through biphasic cells and differential motion detection(pg 153)

4. Object moving stimulus(OMS) ganglion cells have center surround receptive fields with white noise flicker stimulus however comparing the object to the background for motion is what reveals the function of the OMS cells. (159)

5. Retina computations contribute to making predictions for future states with quick reflexive reactions. This is based on the behavioral relevance of approaching objects. If your retina can detect an expanding dark object, such as a bird flying by your face, your body starts to react by flinching or moving away from the approaching object before you actually process what is approaching you. The creation of the reflexive response is based on anticipation. You body reacts to now while your brain is experiencing the past due to sensory delays(155)

6. There is a difference in neural computation if inhibition comes presynaptically or postsynaptically. If there is presynaptic inhibition of bipolar cell terminals they therefore inhibit
the axonal amacrine cells that inhibit the motion detector because these cells detect a difference in background. (154) Object moving stimulus have to postsynaptic inhibition at the bipolar terminals. It has to happen post because at the OMS there is multiple sources of signal combining at the OMS to compute whether an object is moving. If that cells determines there isn’t movement after the cell will inhibit response. (154)

7. Some visual computations needs to be done early in the retina because it takes in a large amount of raw data including all the visual stimulus of the environment. There is a lot of chaos happening in the environment. The early computations start to break down the information and pick out the most relevant information discarding some information before moving on the next levels of computation.