- **Why do we get image aliasing when subsampling and what to do about it?**

- Image aliasing happens when subsampling an image due to the sampling theorem. To capture a signal the sampling frequency needs to be at least twice the hithere are ghest frequency present in the signal. To prevent these, there are anti-aliasing filters, increase the initial sampling rate and even adaptive sampling

- **Why is the notion of scale important in image analysis and computer vision?**

1. Feature detection: Different scales reveral different features of an image. Fine scales show small details while coarse scales highlights larger and more global features. Important for tasks like edge directions, texture analysis and pattern recognition
2. **Robustness to size variation:** Objects in images can appear at various sizes due to distance from the camera or their actual size differences. Analyzing images at multiple scales ensures that algorithms can detect objects regardless of their size in the image
3. **Scale-invariant feature transform (SIFT)**: designed to be scale-invariant meaning they can identify the same features in an image regardless of its scale. Essential for applications like object recognition and image matching

- **What is a scale-space representation? On what basis is it constructed?**

- Method in image processing and computer vision where an image is analyzed at multiple scales or levels of detail. Constructed by progressively smoothing the image, typically using a Gaussian filter with varying standard deviation. This process creates a set of images, each representing the original image at different scales.

    1. Gaussian smoothing: Applying a Gaussian filter to the original image, increasing the blur for higher scales
    2. Scale parameter: The standard deviation of the Gaussian filter determines the scale; the larger values produce more blurred images
    3. Multi-scale analysis: Enables analysis of image features from fine to coarse levels
    4. Feature detection: Useful for detecting scale-invariant features in computer vision algorithms
    5. Consist interpretation: As scale increases, new features are not introduced; existing features either disappear or change position.

- Scale-space representation is crucial for handling varying image sizes and resolutions in image processing tasks.

- **What structural requirements are natural to impose on early visual operations?**

- **Linearity and homogeneity:** Operations should be linear and homogenous to simplify mathematical analysis and ensure that changes in light intensity do not affect the detection of features and structures

- **Scale invariance:** The ability to recognize objects and features regardless of their size in the image is crucial. This requires algorithms that can operate effectively across various scales.
- **Rotation invariance:** The visual system should recognize objects and features regardless of their orientation, requiring operations that are invariant to rotation

- **Spatial locality:** Operations should focus on local spatial information to detect features and patterns. This aligns with the way human vision focuses on small regions of the visual field at a time

- **Robustness to Noise:** Early visual operations should be robust to noise and variations in lightning and contrast, as there are common in real-world imaging conditions.

- **Casualty:** When processing an image over time or across scales, the operations should not introduce features that were not present in the earlier or finer scales

- **Edge preservation:** While smoothing or filtering an image, it's important to preserve edges and boundaries as they are critical for interpreting the structure of objects in the scene

- **Computational Efficiency:** Operations should be designed for computational efficiency allowing for real-time or near-real-time processing which is often essential in practical applications.

  By adhering to these structural requirements, early visual operations in computer vision systems can be more effective in interpreting and analyzing visual data in a manner that is both robust and reflective of human visual perception.

- **What is meant by a Gaussian derivative? Why are these important?**
- They help analyzing images. Useful for tasks for edge detections where you weant to find the places in an image where the colors or intensity change rapidly often indicating the presence of objects or edges.

- **Why is edge detection important for image understanding?**
- Used for object recognition, image segmentation, feature extraction, motion analysis, scene and image enhancement. It is a fundamental step in computer vision because it helps computer see and understand image by identifying

- **What families of methods exist for edge detection?**
- Gradient-based methods
- Zero-crossing detectors
- Laplacians of Gaussians
- Edge enhancement filters
- Morphological filters
- Canny edge detectors
- contour detections

- Deep Learning based methods
- Phase-based methods
- Graph-based methods

- **How does the Canny edge detector work?**
- Works through a series of steps:
1. Gaussian smoothing
2. Gradients calculations
3. Non-maximum suppression → Checks gradient magnitude, filters out all poinst taht cannot be considered edges
4. Edge tracking by hysterioses → Pixels between T_low and T_high.
5. Edge thinning → Find think Edges

- **What is differential edge detection?**
- Basically derive and find edges

- **What should the image derivatives be equal to on edge points?**
- They should be equal to zero but it depends on the gradients. The first gradient just tells us the rate of change. The second gradient tells us if it could be a maximum or minimum and lastly the third one if it is negative then we know for sure that it is an edge.