

---

## Q1. Demonstrate the Use of Homomorphic Filtering

---

### 1. What is Homomorphic Filtering?

Homomorphic Filtering is an **image enhancement technique**.

It is used when an image has **uneven lighting** — for example:

- ✧ One side of the image looks **too dark** (shadow).
- ✧ The other side looks **too bright** (light reflection).

In simple words —

This filter **removes lighting effects** and **makes the image clear and evenly bright**.

---

### 2. Why Do We Need It?

When we take pictures in real life:

- ✧ Light may not fall equally on all parts of the object.
- ✧ Some areas become **too bright** and others **too dark**.

Because of this, important details may get **hidden**.

Homomorphic filtering helps to:

- ✧ **Reduce** the unwanted lighting (illumination)
- ✧ **Increase** the object details (edges, features)

So the final image looks **better and more natural**.

---

### 3. Basic Idea Behind Homomorphic Filtering

Every image can be thought of as made up of **two parts**:

$$f(x, y) = i(x, y) * r(x, y)$$

Where:

$f(x, y)$  ) = Original image

$i(x, y)$  ) = Illumination (light/shadow part)

$r(x, y)$  ) = Reflectance (object details, edges)

In an image:

Illumination changes **slowly** (low-frequency information)

Reflectance changes **fast** (high-frequency information)

So, homomorphic filtering tries to:

**Reduce** the slow part (illumination)

**Enhance** the fast part (details)

---

#### 4. Step-by-Step Working (in Simple Words)

Let's see the process step by step

---

##### Step 1: Start with the image

We have the image:

$$f(x, y) = i(x, y) * r(x, y)$$

Since illumination and reflectance are multiplied, we take **log** to make them easier to handle.

---

##### Step 2: Take the logarithm

$$\ln(f(x, y)) = \ln(i(x, y)) + \ln(r(x, y))$$

Now the multiplication has become addition.  
This helps us separate light and detail parts later.

---

### Step 3: Apply Fourier Transform

We change the image from **spatial domain** (normal image) to **frequency domain** using Fourier Transform.

$$Z(u, v) = F\{\ln(f(x, y))\}$$

Now in this domain:

Illumination → appears in **low frequencies**

Reflectance → appears in **high frequencies**

---

### Step 4: Apply Homomorphic Filter

Now we apply a **special filter** that:

Reduces low frequencies (illumination)

Increases high frequencies (details)

This improves contrast and sharpness.

---

### Step 5: Apply Inverse Fourier Transform

Now we convert the image back to normal view (spatial domain):

$$s(x, y) = F^{-1}\{S(u, v)\}$$

---

### Step 6: Take Exponential

We undo the logarithm by taking exponential:

$$g(x, y) = e^{\{s(x, y)\}}$$

---

Now (  $g(x, y)$  ) is the **final improved image**.

---

## What Happens to the Image?

After applying homomorphic filtering:

- ✧ **Lighting becomes uniform** — dark parts get brighter, bright parts get balanced.
  - ✧ **Details and edges become clearer.**
  - ✧ The whole image looks **sharper and more natural.**
- 

## Real-Life Example

Imagine you clicked a photo of a person standing near a window:

One side of the face is very bright due to sunlight.

The other side is dark because of shadow.

If you apply **Homomorphic Filtering**,  
the filter will **reduce the extra light** and **brighten the dark side** —  
so the whole face becomes evenly visible.

---

---

## . Applications

Field	Use
Photography	Balances light and shadow in photos
Medical Imaging	Improves MRI and X-ray images
Satellite Images	Removes cloud or light variations
Computer Vision	Prepares clear input for recognition systems

---

## . Advantages

- ✓ Removes uneven lighting
  - ✓ Enhances contrast and sharpness
  - ✓ Makes images look natural
  - ✓ Works well on old or dull images
- 

## . Disadvantages

- ✗ Needs careful selection of parameters
  - ✗ If filter is too strong, image can look unnatural
  - ✗ Takes more processing time
- 

Okay Let's explain **Q2: Correlation and Convolution method with 1D example** in **very, very easy and simple English**, like class notes you can write directly in your copy.

---

✧ **Q2. Explain the Correlation and Convolution Method with the Help of a 1D Example**

---

1. Introduction

In image or signal processing, we often use two operations called **Correlation** and **Convolution**.

These are simple mathematical operations that help us to:

- ✧ **Compare** two signals or images (in correlation)
- ✧ **Filter or modify** an image (in convolution)

Both work by **sliding** one signal (or mask) over another signal and **multiplying and adding** the overlapping values.

---

2. What is Correlation?

**Correlation** tells us **how similar** two signals (or images) are.

It is used to **find a pattern or match** inside another signal.

For example:

If you want to find where a small pattern appears inside a big image, you use correlation.

### Simple Meaning:

We keep one signal fixed, and slide the other over it — multiply matching parts and add them.

If the sum is large → signals are similar at that point.

---

### 3. What is Convolution?

**Convolution** is used to **modify or change** a signal or image.

It is used for:

- ✧ **Blurring**
- ✧ **Sharpening**
- ✧ **Edge detection**
- ✧ **Smoothing**

The process is **almost the same** as correlation —  
but before sliding, we **flip the filter** (mask) from left to right.

So convolution = **flip + multiply + add**

---

### 4. Important Difference (in one line)

Operation	What Happens
Correlation	Slide directly — no flip
Convolution	Flip the filter, then slide

---

### 5. Example (1D Signals)

Let's take a small example so you can easily understand.

Given:

Main signal → (  $f = [1, 2, 3]$  )  
Filter → (  $h = [1, 2]$  )

---

#### ↙ (A) Correlation

In correlation, we do **not flip** the filter.

Now slide ( h ) on ( f ) and multiply the matching numbers, then add them.

Step	Overlap	Multiply and Add Result
1	$(1 \times 1) + (2 \times 2)$	$1 + 4 = 5$
2	$(2 \times 1) + (3 \times 2)$	$2 + 6 = 8$

✓ Final answer of **correlation** = [5, 8]

---

## ✓ (B) Convolution

In convolution, we **flip** the filter first.

So, flipped ( h = [2, 1] )

Now slide this flipped filter on ( f ):

Step	Overlap	Multiply and Add Result
1	$(1 \times 2) + (2 \times 1)$	$2 + 2 = 4$
2	$(2 \times 2) + (3 \times 1)$	$4 + 3 = 7$

✓ Final answer of **convolution** = [4, 7]

---

## 6. Easy Way to Remember

Feature	Correlation	Convolution
Filter flipped?	✗ No	✓ Yes
Purpose	Compare / match	Filter / modify
Example use	Template matching	Image blurring or sharpening

## 7. Real-Life Example

### Example 1: Correlation

You have a small image of a face and a big crowd photo.  
You slide the small image over the big one — where the values match best,  
correlation value is highest → **face found**.

### Example 2: Convolution

You use a sharpening filter on a blurry image.  
The convolution operation enhances edges → **image looks clear and sharp.**

---

---

## Q3. Explain the Need of Gray Level Transformation

(In very easy language)

---

### 1. What is Gray Level?

In a **gray-scale image**, every pixel has a value between **0 and 255**.

- ✧ 0 → **Black**
- ✧ 255 → **White**
- ✧ Between 0 and 255 → **Different shades of gray**

Example:

If a pixel = 100 → it is gray.

If a pixel = 200 → it is light gray.

---

### 2. What is Gray Level Transformation?

**Gray Level Transformation** means **changing the brightness and contrast** of an image by changing its pixel values.

It takes an image and applies a simple rule to every pixel —  
to make the image **clearer, brighter, or darker.**

---

### 3. Why Do We Need Gray Level Transformation?

We need it because many images are not perfect.

Sometimes:

- ✧ The image is **too dark**
- ✧ The image is **too bright**
- ✧ The image has **low contrast** (looks dull)

So, we change the gray levels to:

- ✓ Make the dark parts brighter
  - ✓ Make the bright parts normal
  - ✓ Improve contrast and visibility
- 

#### 4. Simple Example

Imagine you took a photo at night.

The image looks too dark, and you can't see anything clearly.

If we apply **gray level transformation**,  
the pixel values will increase —  
dark parts become brighter,  
and the image looks **normal and clear**.

---

#### 5. Main Types of Gray Level Transformations

There are a few common types — let's learn them simply

---

##### (A) Image Negative

This makes **black areas white** and **white areas black**.  
It's like a **photo negative**.

Formula:

$$s = 255 - r$$

Example:

If a pixel = 50 → New pixel =  $255 - 50 = 205$

**Use:** Helpful for medical X-rays or photo films to see details clearly.

---

##### (B) Log Transformation

This makes **dark areas brighter** and **bright areas darker**.

Formula:

$$s = c \log(1 + r)$$

**Use:** When an image has **very dark areas** — like space or satellite images.

---

### (C) Power-Law (Gamma) Transformation

Used to **control brightness**.

Formula:

$$= c * r^{\gamma}$$

If  $\gamma < 1 \rightarrow$  Image becomes brighter

If  $\gamma > 1 \rightarrow$  Image becomes darker

**Use:** In TV, mobile, and computer screens for brightness control.

---

### (D) Contrast Stretching

Used to **improve dull or low-contrast images**.

It spreads pixel values to cover full 0–255 range.

**Use:** To make photos look sharper and clearer.

---

### (E) Thresholding

Converts a **gray image** into a **black and white image**.

Rule:

If  $\text{pixel} \geq T \rightarrow$  make it **white (255)**

If  $\text{pixel} < T \rightarrow$  make it **black (0)**

**Use:** In document scanning and OCR (text recognition).

---

