Name	Adwait Purao
UID no.	2021300101

Experiment 9	
AIM :	Image Processing using Morphological Operation
OBJECTIVE:	1. Develop an enhanced infrared (I-R) and visible (V-I) image fusion framework using morphological operations based unsharp masking for image enhancement. 2. Implement curvelet transform to decompose the source images into detailed and approximation coefficients. 3. Integrate approximation coefficients using the PCA fusion rule and combine detailed coefficients using the max fusion rule. 4. Evaluate the proposed fusion framework against existing methods, assessing both visual quality and metrics values to demonstrate its superiority in image fusion performance.
INTRODUCTION:	The fusion of infrared (I-R) and visible (V-I) images is pivotal in surveillance and remote sensing, enhancing visibility and situational awareness. This paper introduces an advanced fusion framework incorporating a novel enhancement method based on morphological operations and unsharp masking. By employing curvelet transform for coefficient generation and fusion rules like PCA and max fusion, the proposed framework achieves superior image quality and outperforms existing methods. This experiment aims to validate the efficacy of the proposed framework in enhancing image fusion outcomes.

BLOCK DIAGRAM:

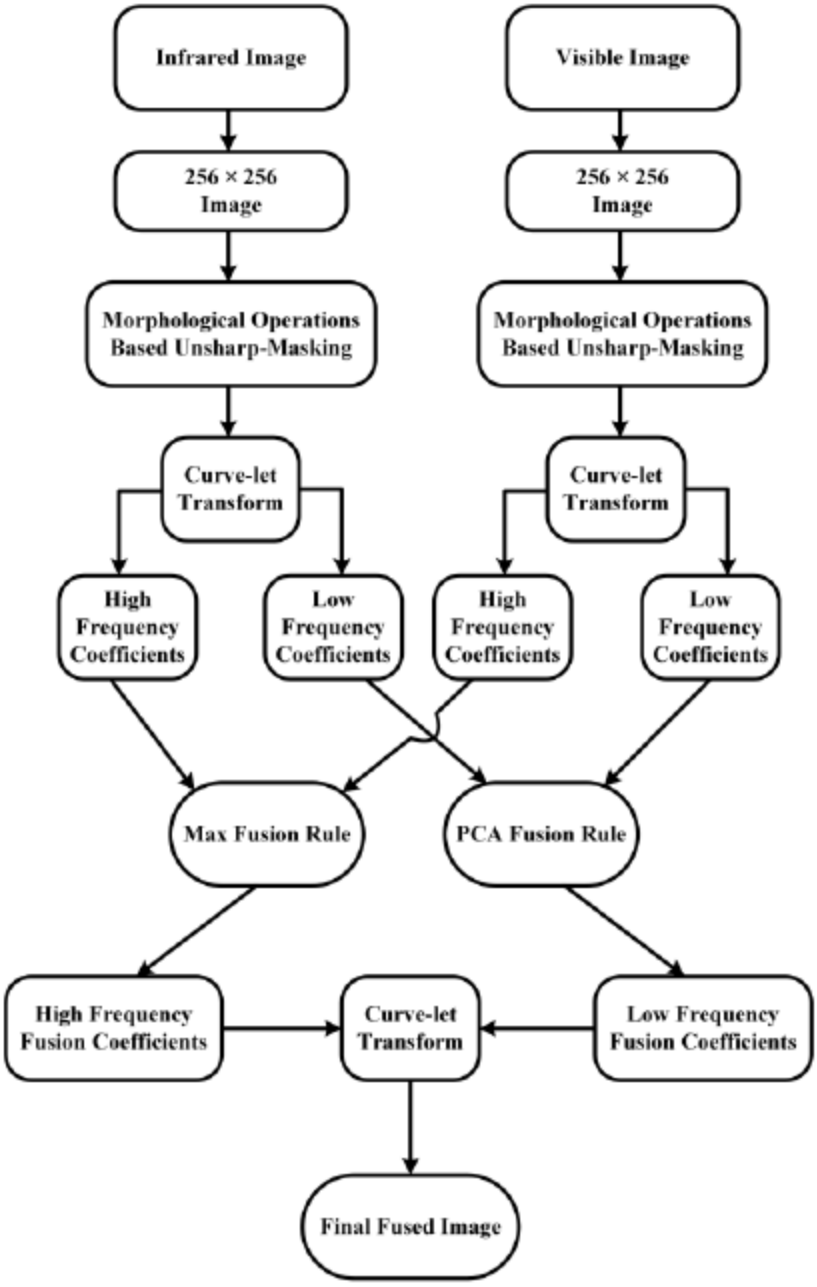


Fig. 1. The block diagram of proposed algorithm.

IMPLEMENTATION:

```
import numpy as np
import cv2
from scipy.ndimage import gaussian_filter
from scipy.ndimage import grey_opening, grey_closing
from pywt import dwt2, idwt2
```

```
from sklearn.decomposition import PCA

def morph_unsharp_mask(img):
    """
    Applies morphological operations based unsharp masking
to enhance the image.
    """
    # Gaussian blurring
    blurred = gaussian_filter(img, sigma=1)

    # Morphological opening
    opened = grey_opening(img, size=(3, 3))

    # Morphological closing
    closed = grey_closing(opened, size=(3, 3))

    # Unsharp masking
    sharpened = img - closed
    enhanced = img + sharpened

    return enhanced

def curvelet_transform(img):
    """
    Applies curvelet transform to decompose the image into
approximation and detailed coefficients.
    """
    cA, (cH, cV, cD) = dwt2(img, 'db1')
    return cA, (cH, cV, cD)

def fusion_approx(cA_ir, cA_vi):
    """
    Fuses the approximation coefficients using PCA.
    """
    X = np.stack([cA_ir.ravel(), cA_vi.ravel()], axis=1)
    pca = PCA(n_components=1)
    cA_fused = pca.fit_transform(X).reshape(cA_ir.shape)
    return cA_fused
```

```
def fusion_detail(cH_ir, cV_ir, cD_ir, cH_vi, cV_vi, cD_vi):
    """
    Fuses the detailed coefficients using the max rule.
    """
    cH_fused = np.maximum(cH_ir, cH_vi)
    cV_fused = np.maximum(cV_ir, cV_vi)
    cD_fused = np.maximum(cD_ir, cD_vi)
    return cH_fused, cV_fused, cD_fused

def fuse_images(ir_img, vi_img):
    """
    Fuses the infrared and visible images using the proposed
method.
    """
    # Resize the images to the same shape
    ir_img = cv2.resize(ir_img, vi_img.shape[:2][::-1])

    # Enhance the source images
    ir_enhanced = morph_unsharp_mask(ir_img)
    vi_enhanced = morph_unsharp_mask(vi_img)

    # Apply curvelet transform
    cA_ir, (cH_ir, cV_ir, cD_ir) =
curvelet_transform(ir_enhanced)
    cA_vi, (cH_vi, cV_vi, cD_vi) =
curvelet_transform(vi_enhanced)

    # Fuse the approximation and detailed coefficients
    cA_fused = fusion_approx(cA_ir, cA_vi)
    cH_fused, cV_fused, cD_fused = fusion_detail(cH_ir,
cV_ir, cD_ir, cH_vi, cV_vi, cD_vi)

    # Reconstruct the fused image
    fused_img = idwt2((cA_fused, (cH_fused, cV_fused,
cD_fused)), 'db1')

    return fused_img

def dwt_sharpen_fusion(ir_img, vi_img):
```

```
    """
    DWT and Sharpen filter based fusion.
    """
    # Resize the images to the same shape
    ir_img = cv2.resize(ir_img, vi_img.shape[:2][::-1])

    # Apply DWT and sharpen filter
    cA_ir, (cH_ir, cV_ir, cD_ir) = dwt2(ir_img, 'db1')
    cA_vi, (cH_vi, cV_vi, cD_vi) = dwt2(vi_img, 'db1')

    cA_fused = (cA_ir + cA_vi) / 2
    cH_fused = np.maximum(cH_ir, cH_vi)
    cV_fused = np.maximum(cV_ir, cV_vi)
    cD_fused = np.maximum(cD_ir, cD_vi)

    dwt_sharpen_result = idwt2((cA_fused, (cH_fused,
cV_fused, cD_fused)), 'db1')
    return dwt_sharpen_result

def pca_multimodal_fusion(ir_img, vi_img):
    """
    PCA based multimodal fusion.
    """
    # Resize the images to the same shape
    ir_img = cv2.resize(ir_img, vi_img.shape[:2][::-1])

    # Apply PCA
    X = np.stack([ir_img.ravel(), vi_img.ravel()], axis=1)
    pca = PCA(n_components=1)
    pca_multimodal_result =
pca.fit_transform(X).reshape(ir_img.shape)
    return pca_multimodal_result

def curvelet_fusion(ir_img, vi_img):
    """
    Curvelet multi-scale transform based fusion.
    """
    # Resize the images to the same shape
    ir_img = cv2.resize(ir_img, vi_img.shape[:2][::-1])
```

```
    # Apply curvelet transform
    cA_ir, (cH_ir, cV_ir, cD_ir) =
curvelet_transform(ir_img)
    cA_vi, (cH_vi, cV_vi, cD_vi) =
curvelet_transform(vi_img)

    cA_fused = fusion_approx(cA_ir, cA_vi)
    cH_fused, cV_fused, cD_fused = fusion_detail(cH_ir,
cV_ir, cD_ir, cH_vi, cV_vi, cD_vi)

    curvelet_result = idwt2((cA_fused, (cH_fused, cV_fused,
cD_fused)), 'db1')
    return curvelet_result

def dwt_unsharp_fusion(ir_img, vi_img):
    """
    DWT and unsharp masking based fusion.
    """
    # Resize the images to the same shape
    ir_img = cv2.resize(ir_img, vi_img.shape[:2][::-1])

    # Apply DWT and unsharp masking
    ir_enhanced = morph_unsharp_mask(ir_img)
    vi_enhanced = morph_unsharp_mask(vi_img)

    cA_ir, (cH_ir, cV_ir, cD_ir) = dwt2(ir_enhanced, 'db1')
    cA_vi, (cH_vi, cV_vi, cD_vi) = dwt2(vi_enhanced, 'db1')

    cA_fused = fusion_approx(cA_ir, cA_vi)
    cH_fused, cV_fused, cD_fused = fusion_detail(cH_ir,
cV_ir, cD_ir, cH_vi, cV_vi, cD_vi)

    dwt_unsharp_result = idwt2((cA_fused, (cH_fused,
cV_fused, cD_fused)), 'db1')
    return dwt_unsharp_result

# Load the input images
```

```
ir_img =
cv2.imread('B:\\morph_processing\\Input_Images\\stairs\\stai
rs_ir.png', cv2.IMREAD_GRAYSCALE)
vi_img =
cv2.imread('B:\\morph_processing\\Input_Images\\stairs\\stai
rs_vi.png', cv2.IMREAD_GRAYSCALE)

# Fuse the images using the different methods
dwt_sharpen_result = dwt_sharpen_fusion(ir_img, vi_img)
pca_multimodal_result = pca_multimodal_fusion(ir_img,
vi_img)
curvelet_result = curvelet_fusion(ir_img, vi_img)
dwt_unsharp_result = dwt_unsharp_fusion(ir_img, vi_img)
proposed_result = fuse_images(ir_img, vi_img)

# Save the fused images
cv2.imwrite('dwt_sharpen_result.png', dwt_sharpen_result)
cv2.imwrite('pca_multimodal_result.png',
pca_multimodal_result)
cv2.imwrite('curvelet_result.png', curvelet_result)
cv2.imwrite('dwt_unsharp_result.png', dwt_unsharp_result)
cv2.imwrite('proposed_result.png', proposed_result)
```

OUTPUT:

Terminal:

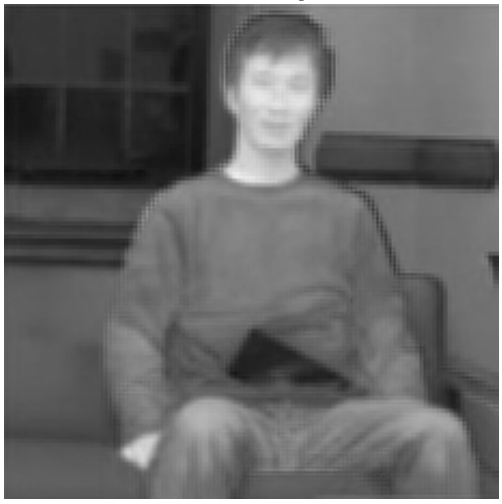
```
PS B:\morph_processing> python .\morph_processing.py
```

Input Images:

VI Image: **IR Image:**

Curvelet:

DWT Sharpen



DWT Unsharp:

PCA Multimodal

	Proposed Result: 
REFERENCE:	S. K. Panguluri and L. Mohan, "An Enhanced Image Fusion Framework Using Morphological Operations Based Unsharp Masking," 2021 International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, India, 2021, pp. 1-6, doi: 10.1109/ICCCI50826.2021.9402531. keywords: {Measurement;Visualization;Surveillance;Transforms;Image fusion;Image reconstruction;Morphological operations;infrared image;visible image;morphological operations based unsharp masking;curve-let transform}, https://ieeexplore.ieee.org/document/9402531

CONCLUSION:

We concluded that the introduced enhanced infrared (I-R) and visible (V-I) image fusion framework represents a significant advancement for surveillance and remote sensing applications. Through the integration of morphological operations and unsharp masking, along with curvelet transform and fusion rules such as PCA and max fusion, the framework produces superior fusion results. Comparative analysis demonstrates its enhanced visual quality and metric values over existing methods, highlighting its potential for enhancing visibility and situational awareness in surveillance scenarios and beyond.