Supplementary Information (SI) for Nanoscale.

Nano-Silver/MoS_2/Nano-Silver Bipolar Photodetector and its Symmetric Ternary-Encoded Image Transmission:Supplementary information

Yangjun Liu, Haifeng Liang, Yang Li, Huang Yuting**

Y.Liu,H.Liang,Y.Li, Y.Huang
Shaanxi Province Thin Film Technology and Optical Test Open Key Laboratory
School of Opto-electronical Engineering
Xi'an Technological University
Xi'an, 710021, China
E-mail:lianghaifeng@xatu.edu.cn

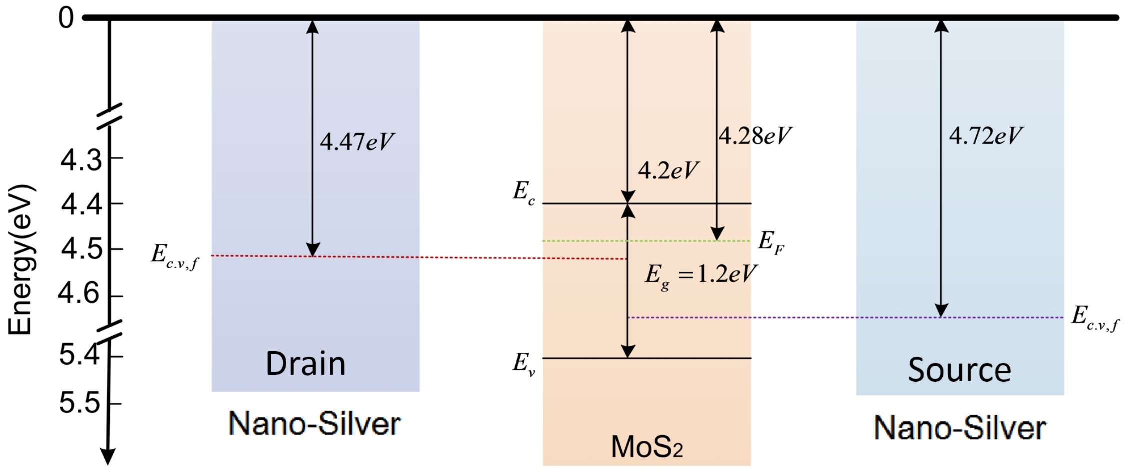

Figure S1: **Energy band diagram profiles of Nano-Silver(Drain), MoS_2, Nano-Silver(Source) prior to contact.** Kelvin probe force microscopy (KPFM) is used to quantitatively determine the energy band alignments of Nano-Silver/MoS_2/ Nano-Silver. The average potential differences between the MoS_2/ Nano-silver(Drain) and MoS_2/ Nano-silver(Source) were measured to be 192 and 445 mV, respectively. Here, the work function values of MoS_2 is 4.28 eV, and the work function values of the Nano-Silver(Drain) and Nano-Silver(Source) can be approximately 4.47 eV and 4.72 eV.

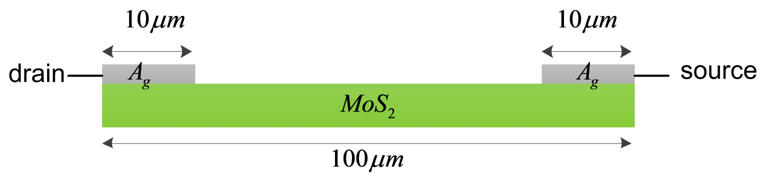

Figure S2: **Geometry of the Nano-silver/MoS_2/Nano-silver device structures considered.** The one-dimensional structure of the detector is shown in Figure S2. The thick of MoS_2 is 30 nm with n-type doping of 4.01×10^{17} cm^{-3}.The vital parameters of the model are shown in Table S1[?].

Table S1: Material and configuration parameters for numerical simulations of device

Material	Nano-silver(drain)	MoS_2	Nano-silver(source)
Temperature(K)	300	300	300
E_g(eV)	0	1.2	0
Work function(eV)	4.47	4.28	4.74
Wavelength(nm)	65	650	650
Doping(cm^{-3})	–	4.01×10^{17} cm^{-3}	–
Lifetime, radiative(ns)	–	1	–
Electronmobility(cm^2/Vs)	–	104	–
m_e^*/m_0	–	0.49	–

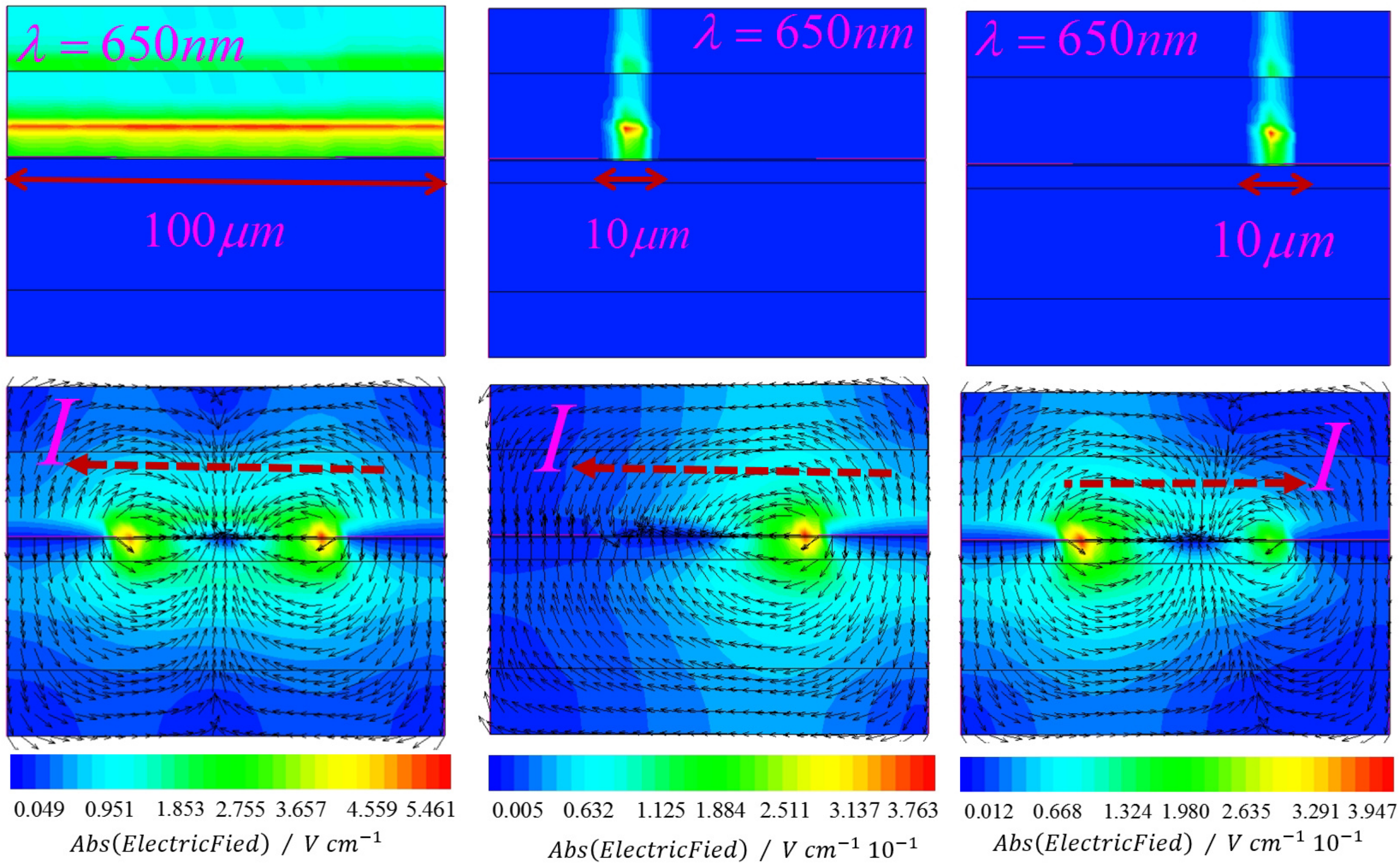

Figure S3: **Numerical simulation of electric field distribution under different positions of illumination.** In the photoelectric generation part, the quantum yield was set to 0.3. The spectral characteristics are calculated using ray tracing when a monochromatic light source was set on MoS_2. And the n and k values of MoS_2 are taken from the test data in the reference[?]. The optical window length is set to 10 µm and placed in different positions. The simulation results show that when the incident light source is placed in the Source region, the photo generated potential decreases the contact potential, and the polarity of the current changes.

Figure S4: **The input image displaying the encoded digital 0/1 matrix.** Each pixel in a grayscale image is represented by a grayscale value of [0 255], and the computer storage of grayscale values requires converting decimal to binary, consisting of 0 and 1. Preprocess images using Python, including grayscale conversion, normalization.

- **grayscale conversion code**

  ```
  import numpy as np
  np.set_printoptions(threshold=5000)
  import matplotlib.pyplot as plt
  image = plt.imread('/home/lyjyy/s/image.jpg')
  plt.imshow(gray_matrix, cmap ='gray')
  ```

- **normalization code**

  ```
  from PIL import Image
  pixels = np.array(gray_matrix)/255.0
  symmetric_pixels = 2 * pixels - 1
  ternary = np.zeros((64, 64), dtype=int)
  ternary = symmetric_pixels
  print(ternary)
  ```

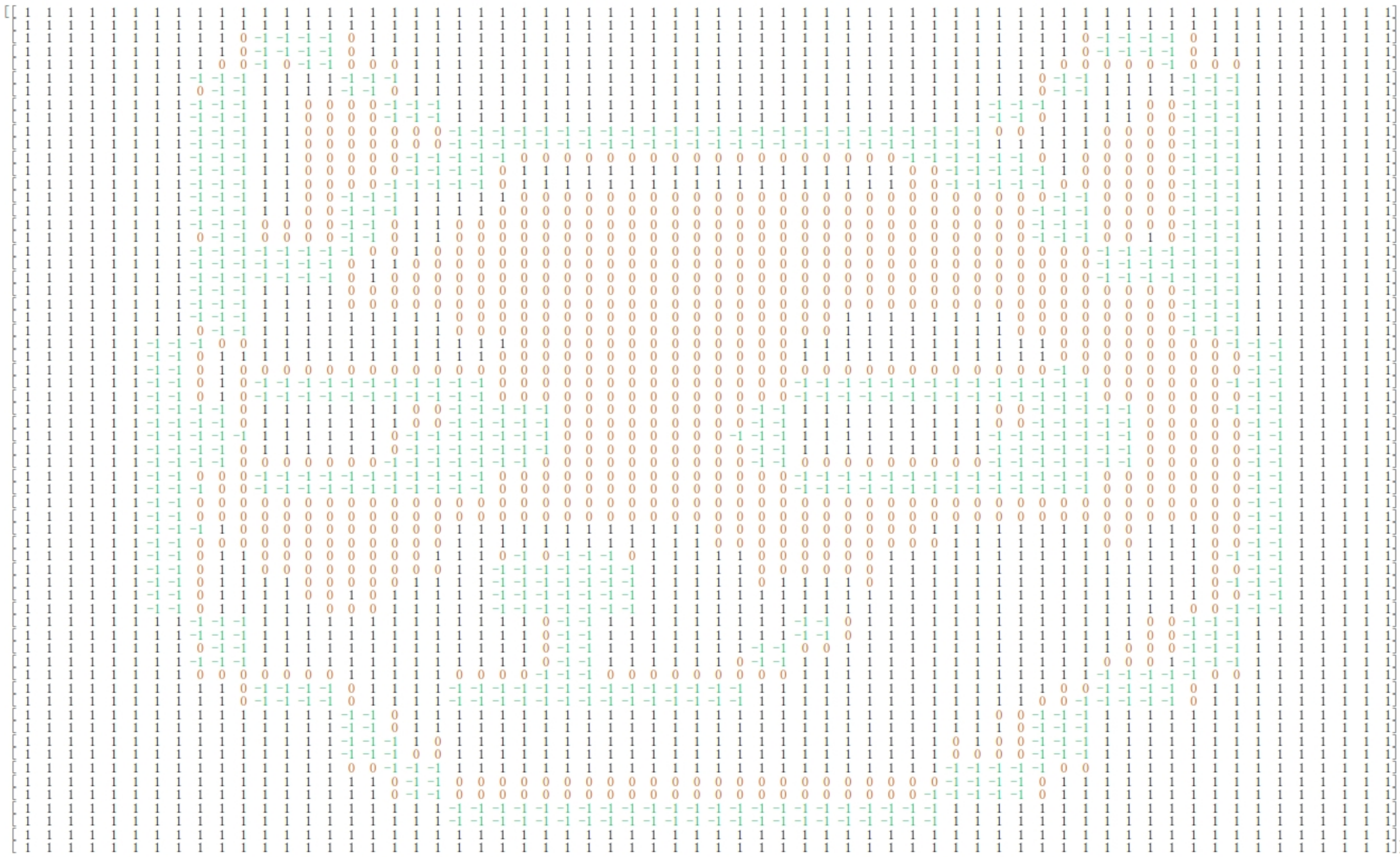

Figure S5: **Symmetrical ternary coding of normalized image.** Use two thresholds for three value quantization. Due to symmetry, we can choose symmetric thresholds such as [-0.33, 0.33] as the interval of 0, with values less than -0.33 being -1 and values greater than 0.33 being 1.

- **matrix ternary transformation code**

```
from PIL import Image
pixels = np.array(gray_matrix) / 255.0
symmetric_pixels = 2 * pixels - 1
ternary = np.zeros((64, 64), dtype=int)
ternary[symmetric_pixels > 0.33] = 1
ternary[symmetric_pixels < -0.33] = -1
print(ternary)
```

Figure S6: **The image is reconstructed by decoding the ternary encoded image information.**

- **Decoding and reconstructing images code**

```
reconstructed = ternary.astype(float)
reconstructed[ternary == 1] = 1.0
reconstructed[ternary == -1] = -1.0
reconstructed[ternary == 0] = 0.0
print(reconstructed)
output_img = Image.fromarray(((reconstructed + 1) * 127.5).astype(np.uint8))
print(output_img)
plt.imshow(output_img, cmap ='gray')
```

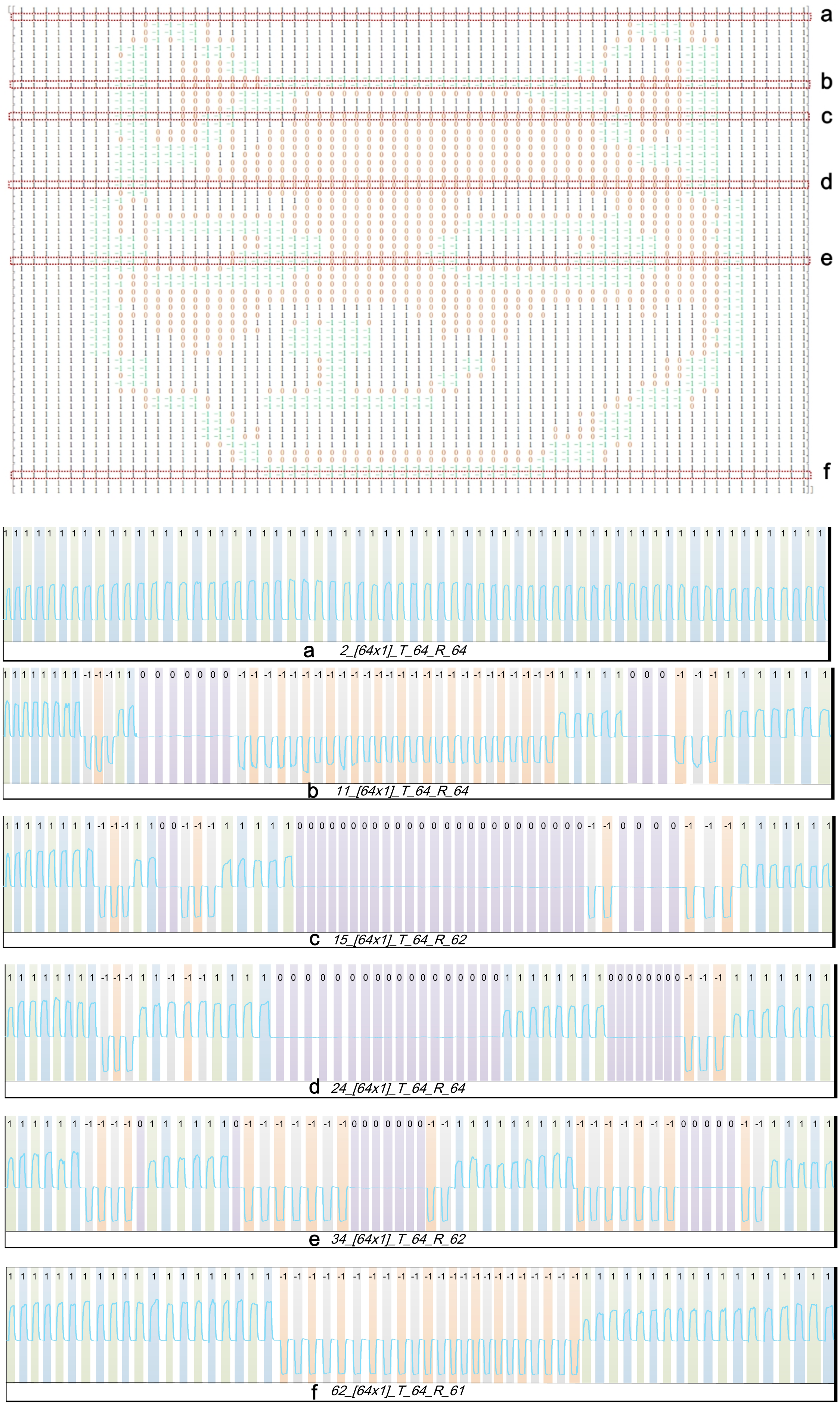

Figure S7: **Triple encoding of images.**

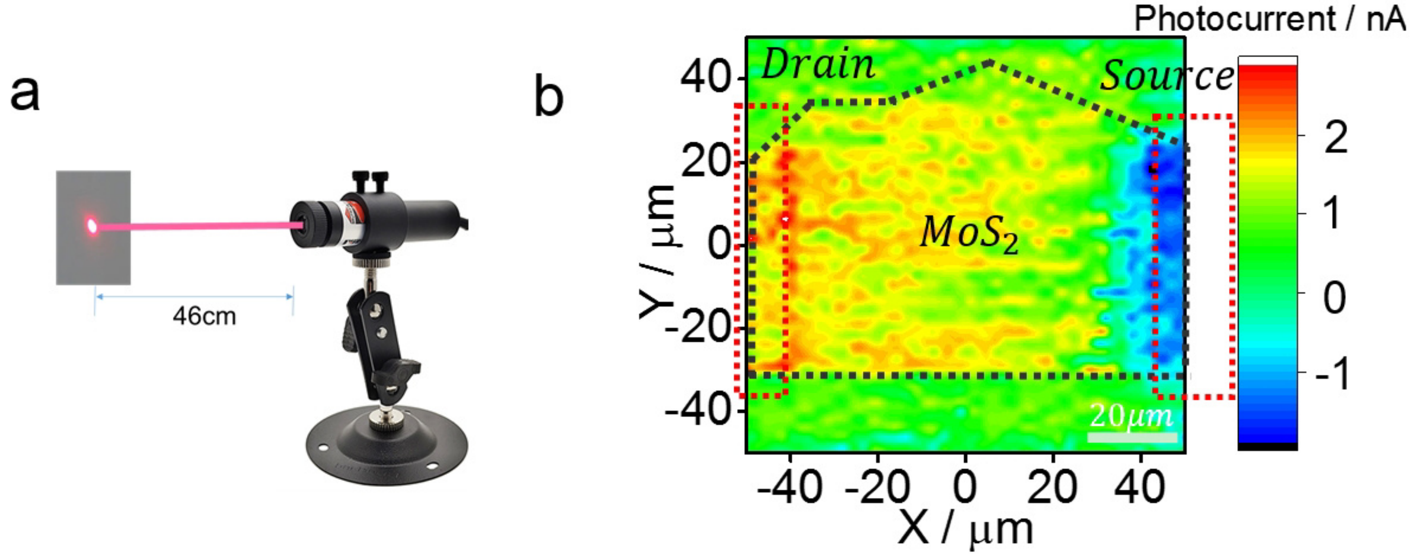

Figure S8: **Schematic diagram of device testing. a) A miniature semiconductor laser for experimental use with a 4 mm spot size and 650 nm wavelength. b) Spatial Photocurrent mapping of the device under 650 nm light illumination at 0 V bias voltage (Mean = 0.94±0.27 nA, n=10000).**

Supplementary Note1. Calculation of the shockley equation fitting curve.

We use the Shockley equation to describe the current voltage relationship of the ideal Schottky junction[?]. The form of Shockley equation is:

$$I = I_0 \left[\exp\left(\frac{qV}{nkT} \right) - 1 \right] \tag{1}$$

where:

- I is device current.
- I_0 is reverse saturation current.
- q is Electron charge, $q = 1.6 \times 10^{-19} C$
- V is bias voltage.
- n is ideal factor.
- k is boltzmann constant, $k = 1.38 \times 10^{-23} J/K$.
- T is absolute temperature,$T = 300K$.

The reverse saturation current equation is :

$$I_0 = AA^* T^2 \exp\left(-\frac{q\Phi_B}{kT} \right) \tag{2}$$

where:

- A is area of junction.
- A^* is effective richardson constant,$A^* = 4\pi q m^* k^2/\hbar^3 \approx 4.4 \times 10^{-4} A/cm^2K^2$.
- Φ_B is Schottky barrier height.

The formula is simplified to:

$$\ln I = \ln\left(AA^*T^2\right) - \frac{q\Phi_B}{kT} + \ln\left[\exp\left(\frac{qV}{nkT} \right) - 1 \right] \tag{3}$$

When the bias voltage is large, $\exp\left(\frac{qV}{nkT}\right) \gg 1$.

$$\ln I = \left[\ln\left(AA^*T^2\right) - \frac{q\varphi}{kT} \right] + \frac{qV}{nkT} \tag{4}$$

The ideal factor is related to the slope of the straight line, and the barrier height is related to the longitudinal intercept $(\ln I_F\,(V=0) = \ln I_0$, that is:

$$\begin{cases} n = \frac{q}{kT}\frac{dV}{d\ln I_F} \\ \Phi_B = \frac{kT}{q} In\left(\frac{A^*T^2}{I_0}\right) \end{cases} \tag{5}$$

The fitting slope value of IV curve is 6.77($\frac{dV}{dLnI_F} = \frac{nKT}{q} = 6.77$), The longitudinal intercept value is -22.64($Ln\,(I_0) = -22.64$).The ideal factor value is calculated as 5.7142, and the barrier height is calculated as 0.42 eV.

Supplementary Note2: Calculation of photo generated carrier concentration.

The incident light power density is calculated as:

$$P_0 = \pi(d/2)^2 = 0.03328W/cm^2 \tag{6}$$

where:

- λ is incident wavelength, $\lambda = 650nm$.
- P_0 is incident light power, $P = 4.18mW$.
- d is spot diameter, $d = 0.4cm$.

The photon energy is calculated as :

$$E = \frac{hc}{\lambda} \approx 3.06 \times 10^{-19}J \tag{7}$$

where:

- h is planck constant, $h = 6.625 \times 10^{-34}J \cdot s$.
- c is speed of light, $c = 3 \times 10^8 m \cdot s^{-1}$.

Carrier generation rate per unit area:

$$G_s = \frac{P_0 \cdot A \cdot \eta}{hc/\lambda} = 9.79 \times 10^{15}cm^{-2}s^{-1} \tag{8}$$

where:

- A is Light absorption rate of MoS_2, $A = 0.3$.
- η is EQE of MoS_2, $\eta = 0.3$.

Unit volume generation rate:

$$G = \frac{G_s}{d} = 3.263 \times 10^{21}cm^{-3}s^{-1} \tag{9}$$

where:

- d is thickness of MoS_2, $d = 30nm$.

Under the condition that the generation rate is equal to the recombination rate, the concentration of photo generated carriers in the device in steady state is:

$$\Delta n = G \cdot \tau = 3.263 \times 10^{21}cm^{-3}s^{-1} \cdot 10^{-9}s = 3.263 \times 10^{12}cm^{-3} \tag{10}$$

where:

- τ is thickness of MoS_2, $\tau = 1ns$.

Supplementary Note3: Calculation of Peak Signal-to-Noise-Ratio(PSNR).

PSNR[?] reflects the ratio between the signal (source image) and the noise (distorted part). It is mainly used to evaluate the effectiveness of image compression, transmission, or reconstruction algorithms. A higher PSNR value indicates less noise and higher image quality. PSNR is defined as:

$$PSRN = 10\log_{10}\left(\frac{MAX^2}{MSE}\right) \tag{11}$$

where:

- MES is the average difference in pixel values between two images.

$$MSE = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}\left[I_1\left(i,j\right) - I_2\left(i,j\right)\right]^2$$

- I_1 is source image, I_2 is reconstructed image.
- M is the height, N is the width.
- i, j are Pixel position index.
- MAX is the maximum possible pixel value in the image.
- **PSNR code**

```
import cv2
import numpy as np
def calculate_psnr(image1_path, image2_path):
img1=cv2.imread(image1_path)
img2=cv2.imread(image2_path)
if img1.shape != img2.shape:
raise ValueError("Error")
mse=np.mean((img1-img2)**2)
if mse == 0:
return float('inf')
PIXEL_MAX=255.0
psnr=10*np.log10((PIXEL_MAX **2)/mse)
return psnr
psnr_value=calculate_psnr('/home/lyjyy/s/structed3.png','/home/lyjyy/s/reconstructed.png')
print(f"PSNR_value:psnr_value:.2f dB")
PSNR_value:29.04 dB
```

Supplementary Note4: Calculation of Structural Similarity Index(SSIM).

SSIM[?] is a perceptual model based on the human visual system (HVS), which is an indicator used to measure the similarity between two images in terms of brightness, contrast, and structure. The core idea of SSIM is to view images as a set composed of brightness, contrast, and structure, and evaluate overall similarity by comparing the similarities in these three aspects. And is given by:

$$SSIM(x,y)=\frac{\left(2\mu_x\mu_y+C_1\right)\left(2\sigma_{xy}+C_2\right)}{\left(\mu_x^2+\mu_y^2+C_1\right)\left(\sigma_x^2+\sigma_y^2+C_2\right)} \tag{12}$$

where:

- $\mu_x, \mu_y$are the average values of images x and y within the local window, description for brightness.
- σ_x^2, σ_y^2 are the variance values of images x and y within the local window, description for contrast.
- σ_{xy} is the covariance values of images x and y within the local windowdescription for structure.
- MAX is the maximum possible pixel value in the image.
- $C_1 = (K_1 L)^2, C_2 = (K_2 L)^2$.
- $K_1 = 0.01, K_2 = 0.03, L = 255$.
- **SSIM code**

```
import sys
sys.argv = ['run.py']
import cv2
import numpy as np
from scipy.ndimage import gaussian_filter
def load_image_grayscale(image_path):
img = cv2.imread(image_path, cv2.IMREAD_GRAYSCALE)
if img is None: raise ValueError(f " Unable to read image: image_path")
img = img.astype(np.float64) / 255.0
return img
def compute_statistics(img1, img2, window_size=11, sigma=1.5):
mu1 = gaussian_filter(img1, sigma=sigma)
mu2 = gaussian_filter(img2, sigma=sigma)
sigma1_sq = gaussian_filter(img1 ** 2, sigma=sigma) - mu1 ** 2
sigma2_sq = gaussian_filter(img2 ** 2, sigma=sigma) - mu2 ** 2
sigma12 = gaussian_filter(img1 * img2, sigma=sigma) - mu1 * mu2
return mu1, mu2, sigma1_sq, sigma2_sq, sigma12
def calculate_ssim(img1, img2, window_size=11, sigma=1.5, K1=0.01, K2=0.03, L=1.0):
mu1, mu2, sigma1_sq, sigma2_sq, sigma12 = compute_statistics(img1, img2, window_size, sigma)
C1 = (K1 * L) ** 2
C2 = (K2 * L) ** 2
luminance = (2 * mu1 * mu2 + C1) / (mu1 ** 2 + mu2 ** 2 + C1)
contrast = (2 * np.sqrt(np.abs(sigma1_sq)) * np.sqrt(np.abs(sigma2_sq)) + C2) / (sigma1_sq +
sigma2_sq + C2)
structure = (sigma12 + C2 / 2) / (np.sqrt(np.abs(sigma1_sq)) * np.sqrt(np.abs(sigma2_sq)) + C2
/ 2)
ssim_map = luminance * contrast * structure
return np.mean(ssim_map)
def ssim_index(image1_path, image2_path, window_size=11, sigma=1.5, K1=0.01, K2=0.03):
img1 = load_image_grayscale(image1_path)
```

```
img2 = load_image_grayscale(image2_path)
if img1.shape != img2.shape: raise ValueError(”The two images you enter must have the same size
”)
ssim = calculate_ssim(img1, img2, window_size, sigma, K1, K2, L=1.0)
return ssim
if _name_ == ”_main_”:
image1_path = ’/home/lyjyy/s/structed3.png’
image2_path = ’/home/lyjyy/s/reconstructed.png’
try:
ssim_value = ssim_index(image1_path, image2_path)
print(f ”SSIM value is: ssim_value:.4f”)
except Exception as e:
print(f ”Error: e”)
SSIM value is: 0.9060
```

Supplementary Note5: Calculation of noise-equivalent power(NEP).

The transit-time bandwidth f_{tr} of the device was estimated to be approximately 300 GHz. And is given by:

$$f_{tr} \approx \frac{0.45 v_{sat}}{W} \tag{13}$$

Where:

- v_{sat} is the saturation drift velocity.
- W is the depletion region width.

The total capacitance C_T was calculated as:

$$C_T = \frac{\varepsilon_0 \varepsilon_r L W}{d} \tag{14}$$

Where:

- ϵ_0 is Vacuum dielectric constant, $\epsilon_0 = 8.85 \times 10^{12} F/m$.
- ϵ_r is relative permittivity of MoS2, $\varepsilon_r \approx 4$.

The calculated RC-time constant bandwidth f_{rc} equation is :

$$f_{rc} = \frac{1}{2\pi R_T C_T} \tag{15}$$

Where:

- R_T was measured to be $9K\Omega$.
- C_T was calculated to be 4.72 pF.

Then ,the calculated RC-time constant bandwidth f_{rc} was 3.75 MHz. and the total bandwidth f_{-3dB} of the device was calculated to be 3.75 MHz.

$$f_{-3dB} \approx \frac{1}{\sqrt{\left(\frac{1}{f_{tr}}\right)^2 + \left(\frac{1}{f_{rc}}\right)^2}} \tag{16}$$

Although the device has a high transit-time bandwidth, the RC time constant bandwidth is the primary limiting factor. Based on the calculation results, the device falls into the mid-frequency application range.Furthermore, we calculated the thermal noise $i_thermal$(Johnson-Nyquist noise) , the dark current noise i_shot(shot noise), and 1/f noise to be $2.63nA$, $9.8pA$, and $1.31 \times 10^{-20}A$, respectively.

$$i_{thermal} = \sqrt{\frac{4k_B T \Delta f}{R}} \tag{17}$$

$$i_{shot} = \sqrt{2qI_d \Delta f} \tag{18}$$

$$\left\langle i_{1/f}^2 \right\rangle = \int_{10}^{3750000} S_i\,(f) df \tag{19}$$

Thermal noise is the dominant source of noise in the device, while the influence of 1/f noise is minimal and can be neglected. At this point, The noise equivalent power (NEP) was determined to be $2.27 \times 10^{-6} W/\sqrt{Hz}$.

$$NEP = \frac{\sqrt{i_{thermal}^2 + i_{shot}^2}/\sqrt{\Delta f}}{R} \tag{20}$$

Furthermore, the theoretical rise time of the device was calculated to be $0.1\mu S$, which is significantly lower than the experimentally measured value of 3S. The capacitance of the device, estimated from the actual rise time $\tau \approx 3S$, is approximately 152 μF—a value which is clearly unattainable for small-scale semiconductor devices. Therefore, the long rise time of the device is attributed to the presence of trap energy levels on the material surface. Charge carriers captured by these traps require a considerably long time to be released, resulting in a slow increase of current over time.
The signal-to-noise ratio (SNR) of the device can be expressed as:

$$SNR = \frac{P_{in}}{NEP * \sqrt{\Delta f}} \tag{21}$$

Peak Signal-to-Noise Ratio (PSNR) is a metric for image quality. In a system dominated by additive white Gaussian noise, the mean square error of the pixel values is approximately equal to the variance of the noise. The relationship between them is given by:

$$PSNR \propto SNR \tag{22}$$

meaning that for every 1 dB decrease in SNR, the PSNR of the reconstructed image decreases by approximately 1 dB. Furthermore, noise corrupts the local structure and texture of the image, leading to a loss of pixel correlation, which consequently reduces the Structural Similarity (SSIM) index of the reconstructed image. Their relationship is described by:

$$SSIM \propto SNR \tag{23}$$

Therefore, a high SNR is beneficial for the quality of the reconstructed image.