Lab 1

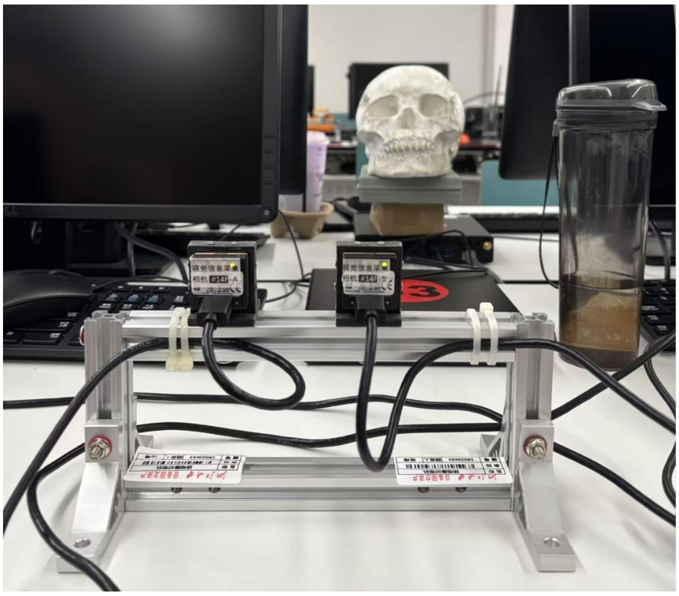
1. Basic Part

1.1 Introduction

The purpose of this experiment is to investigate how key camera parameters—specifically exposure time, gain, and gamma—affect the brightness, contrast, and noise characteristics of captured images. By systematically varying these parameters, we can better understand their roles in image formation and evaluate the trade-offs between brightness, detail preservation, and noise. Such knowledge is essential for optimizing imaging performance in practical applications, particularly under different lighting conditions.

1.2 Experiment Setup

Equipment: Firefly FFY-U3-16S2M camera



1.3 Result and Data Processing

(1) **Exposure time** series: Gain = 0 dB, Gamma = 0.8; exposure time ranged from 9985 µs to 30236 µs.

Exposure Time	Picture
9985 us	

13170 us	
19955 us	
22918 us	
30236 us	

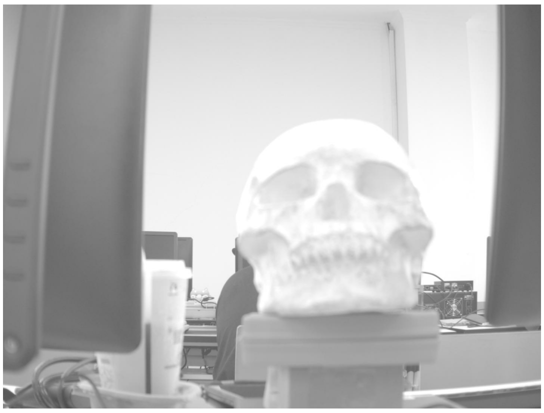
(2) **Gain** series: Exposure time = 19955 μs, Gamma = 0.8; gain increased from 0 dB

to 5 dB.

Gain	Picture
0 db	
1db	
2 db	
3 db	

4db	
5 db	

(3) **Gamma** series: Exposure time = 19955 μs, Gain = 0 dB; gamma adjusted from 0.25 to 1.25.

Gamma	Picture
0.25	

0.6	
0.8	
1.0	
1.25	

1.4 Analysis and Discussion

Exposure Time

Increasing exposure time allows the sensor to capture more photons, which enhances overall image brightness and reduces the relative impact of noise. However, when the exposure time is excessively long (e.g., 30,236 μs), bright regions exceed the sensor's dynamic range, leading to highlight overexposure and loss of detail.

Gain

With exposure time and gamma held constant, increasing gain raises image brightness but also amplifies noise. As a result, the image becomes noticeably grainier, and fine details are degraded.

Gamma

Adjusting gamma alters the brightness distribution without changing the amount of captured signal.

(1) When **Gamma < 1**, shadow regions are brightened, improving visibility of dark details, while highlights are compressed and contrast is reduced.
(2) When **Gamma = 1**, the image maintains its original, linear contrast.
(3) When **Gamma > 1**, highlights are enhanced and shadows are darkened, producing a higher-contrast image but with a loss of detail in dark areas.

1.5 Conclusion

The experiments demonstrate that different imaging parameters influence image quality in distinct ways.

Overall, the results indicate that exposure time primarily controls photon capture and signal quality, gain adjusts brightness at the cost of increased noise, and gamma modifies the perceptual distribution of brightness and contrast.

2. Bonus

2.1 Introduction

Image sensor noise characterization is a fundamental aspect of digital imaging systems that directly impacts image quality, measurement accuracy, and subsequent computer vision applications. Understanding the quantitative relationship between sensor parameters and noise behavior is crucial for optimizing camera settings and developing effective denoising algorithms.

This experiment aims to **establish a quantitative relationship model between noise and key parameters** by systematically controlling the exposure time and gain of an image sensor and acquiring multiple sets of image data, thereby enabling an in-depth understanding of image sensor noise characteristics.

2.2 Experimental Setup and Methodology

2.2.1 Parameter Configuration

Parameter Type	**Values**
Exposure Time (μs)	500, 1000, 5000, 10000, 15000, 20000
Gain (dB)	0.0, 5.0, 10.0, 13.0, 15.0, 17.0, 20.0
Gamma Value	1.0 (Fixed)
Images per Setting	5

2.2.2 Data Acquisition

Image acquisition was automated using a custom Python script based on the Spinnaker SDK (**grab_spinnaker_param_sweep.py**). The script iterated through all parameter combinations, configured the camera settings, captured the specified number of images per setting, and saved them along with a comprehensive metadata file (**metadata.csv/metadata.json**) recording all parameters for each image.

2.3 Quantitative Noise Analysis

2.3.1 Multi-frame Denoising

For each parameter set, the five captured images were averaged to create a "ground truth" denoised image, providing a robust estimate of the underlying signal.

```python
def multi_frame_denoising(image_paths):
    frames = []
    for path in image_paths:
        img = cv2.imread(str(path))
        img_rgb = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
        frames.append(img_rgb.astype(np.float32) / 255.0)

    # 多帧平均去噪
    denoised_image = np.mean(frames, axis=0)
    denoised_image = np.clip(denoised_image, 0, 1)

    return denoised_image, frames
```

2.3.2 Calculate Noise Properties

The noise map for each individual image was calculated as the pixel-wise difference between the original image and the denoised reference:

$$\text{Noise} = \text{Original Image} - \text{Denoised Image}$$

This method preserves the sign of the noise, which is crucial for accurate statistical analysis.

```python
def calculate_noise_properties(frames, denoised):
    # 计算噪声的多个属性，包括均值、标准差和空间分布。
    noise_maps = []
    for i, frame in enumerate(frames):
        noise = frame - denoised
        noise_maps.append(noise)
        print(f"已计算第{i + 1}张图像的原始噪声（保留符号）")

    # 1. 计算平均噪声矩阵
    avg_noise_matrix = np.mean(noise_maps, axis=0)

    # 2. 计算用于可视化的噪声分布图
    avg_noise_abs = np.mean(np.abs(noise_maps), axis=0)
    avg_noise_visual = (avg_noise_abs - np.min(avg_noise_abs)) / (np.max(avg_noise_abs) - np.min(avg_noise_abs) + 1e-8)

    # 3. 返回原始噪声图列表，用于计算整体标准差
    return avg_noise_matrix, avg_noise_visual, noise_maps
```

2.3.3 Relationship Between Noise and Exposure Time

With the gain fixed at 5.0, the relationship between noise standard deviation and exposure time was fitted using a quadratic model:

$$\text{noise} = a \times \text{exp}^2 + b \times \text{exp} + c$$

This choice of model stems from the composite nature of image sensor noise, which primarily originates from three independent sources with distinct dependencies on exposure time:

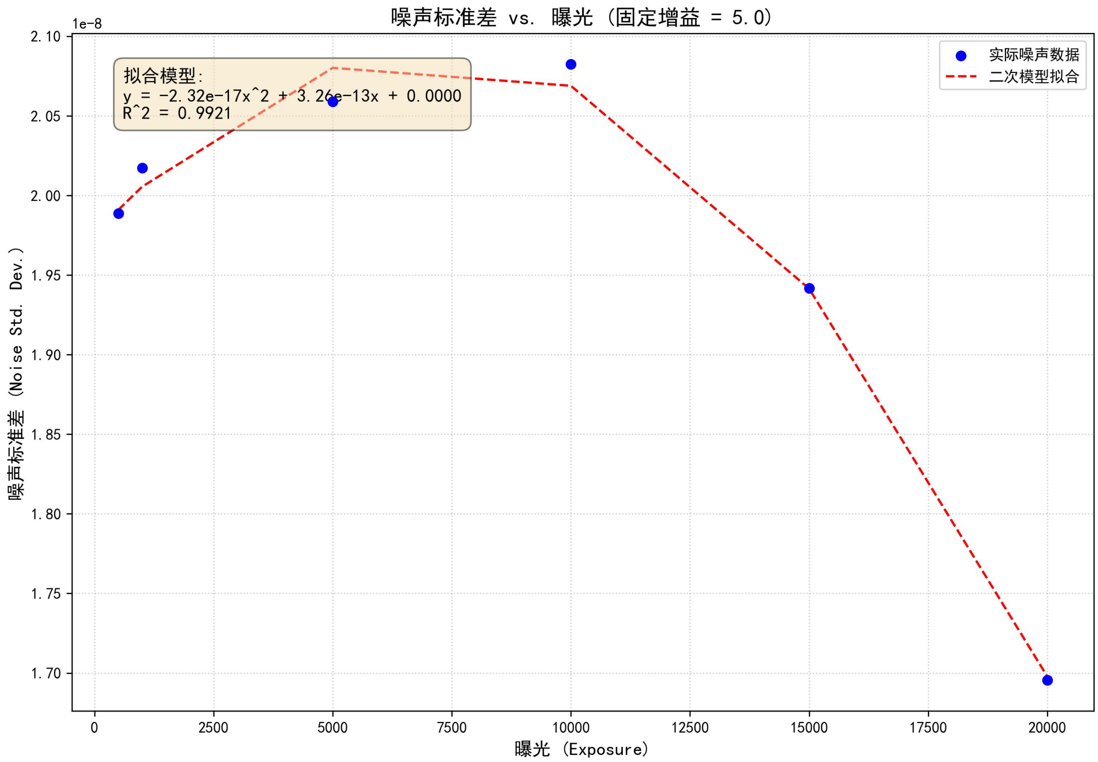
(1) Photon Shot Noise: $\sigma_{\text{shot}} \propto \sqrt{\text{exp}}$

(2) Dark Current Noise: $\sigma_{\text{dark}} \propto \sqrt{\text{exp}}$

(3) Read Noise: $\sigma_{\text{read}} = \text{constant}$

Since these noise sources are statistically independent, their contributions to the total noise follow the principle of variance superposition. The total noise variance is the sum of the variances of each component. The quadratic model effectively captures this composite non-linearity, making it an appropriate choice for fitting the experimental data.

```
def fit_and_plot_Exposure(result_root_dir, noise_data, fixed_gain_for_exp_analysis):
    # 1. 定义二次拟合模型
    def quadratic_model(x, a, b, c):
        return a * x**2 + b * x + c

    # 2. 根据固定增益筛选数据，按曝光时间升序排序
    exp_data = sorted(
        [d for d in noise_data if d['gain'] == fixed_gain_for_exp_analysis],
        key=lambda x: x['exp']
        )

        # 3. 提取自变量和因变量
        exposures = np.array([d['exp'] for d in exp_data])
        noise_stds_exp = np.array([d['noise_std'] for d in exp_data])

        # 4. 二次模型拟合，计算拟合优度
        try:
            params_quad, _ = curve_fit(quadratic_model, exposures, noise_stds_exp)
            y_fit_quad = quadratic_model(exposures, *params_quad)
            r2_quad = r2_score(noise_stds_exp, y_fit_quad)
```



$$y = -2.32 \times 10^{-17} x^2 + 3.26 \times 10^{-13} x$$

$$R^2 = 0.9921$$

2.3.4 Relationship Between Noise and Gain

With the exposure time fixed at 5000μs, a comparative fitting was performed using multiple models:

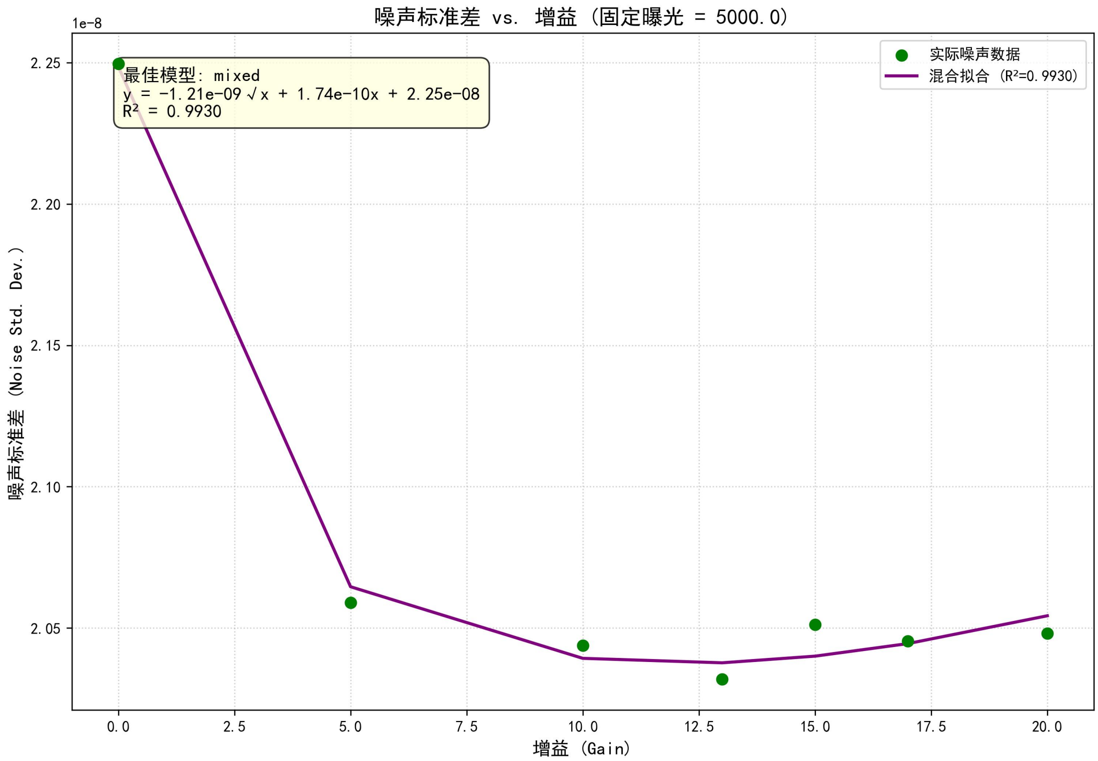
(1) Linear Model: noise=a×gain+b

(2) Square Root Model: noise=$a\times\sqrt{\text{gain}}+b$

(3) Mixed Model: noise=$a\times\sqrt{\text{gain}}+b\times\text{gain}+c$

```
def fit_and_plot_Gain(result_root_dir, noise_data, fixed_exp_for_gain_analysis):
    # 1. 定义三种候选模型
    def linear_model(x, a, b):
        return a * x + b
```

```python
def sqrt_model(x, a, b):
    return a * np.sqrt(x) + b
def mixed_model(x, a, b, c):
    return a * np.sqrt(x) + b * x + c

    # 2. 根据固定曝光筛选数据，按增益升序排序
    gain_data = sorted(
        [d for d in noise_data if d['exp'] == fixed_exp_for_gain_analysis],
        key=lambda x: x['gain']
    )

    # 3. 提取自变量和因变量
    gains = np.array([d['gain'] for d in gain_data])
    noise_stds_gain = np.array([d['noise_std'] for d in gain_data])

    # 4. 遍历三种模型，逐一拟合并评估，对比更新最优模型
    best_r2 = -np.inf
    best_model_name = None
    best_params = None
    best_y_fit = None

    models = {'linear': linear_model, 'sqrt': sqrt_model, 'mixed': mixed_model}

    for model_name, model_func in models.items():
        try:
            params, _ = curve_fit(model_func, gains, noise_stds_gain)
            y_fit = model_func(gains, *params)
            r2 = r2_score(noise_stds_gain, y_fit)

            if r2 > best_r2:
                best_r2 = r2
                best_model_name = model_name
                best_params = params
                best_y_fit = y_fit

            print(f"{model_name}模型 R²: {r2:.4f}")

        except RuntimeError:
            print(f"无法拟合 {model_name} 模型")
```



$$y = -1.21 \times 10^{-9}\sqrt{x} + 1.74 \times 10^{-10}x + 2.25 \times 10^{-8}$$

$$R^2 = 0.9930$$

2.3.5 Three-Dimensional Noise Model

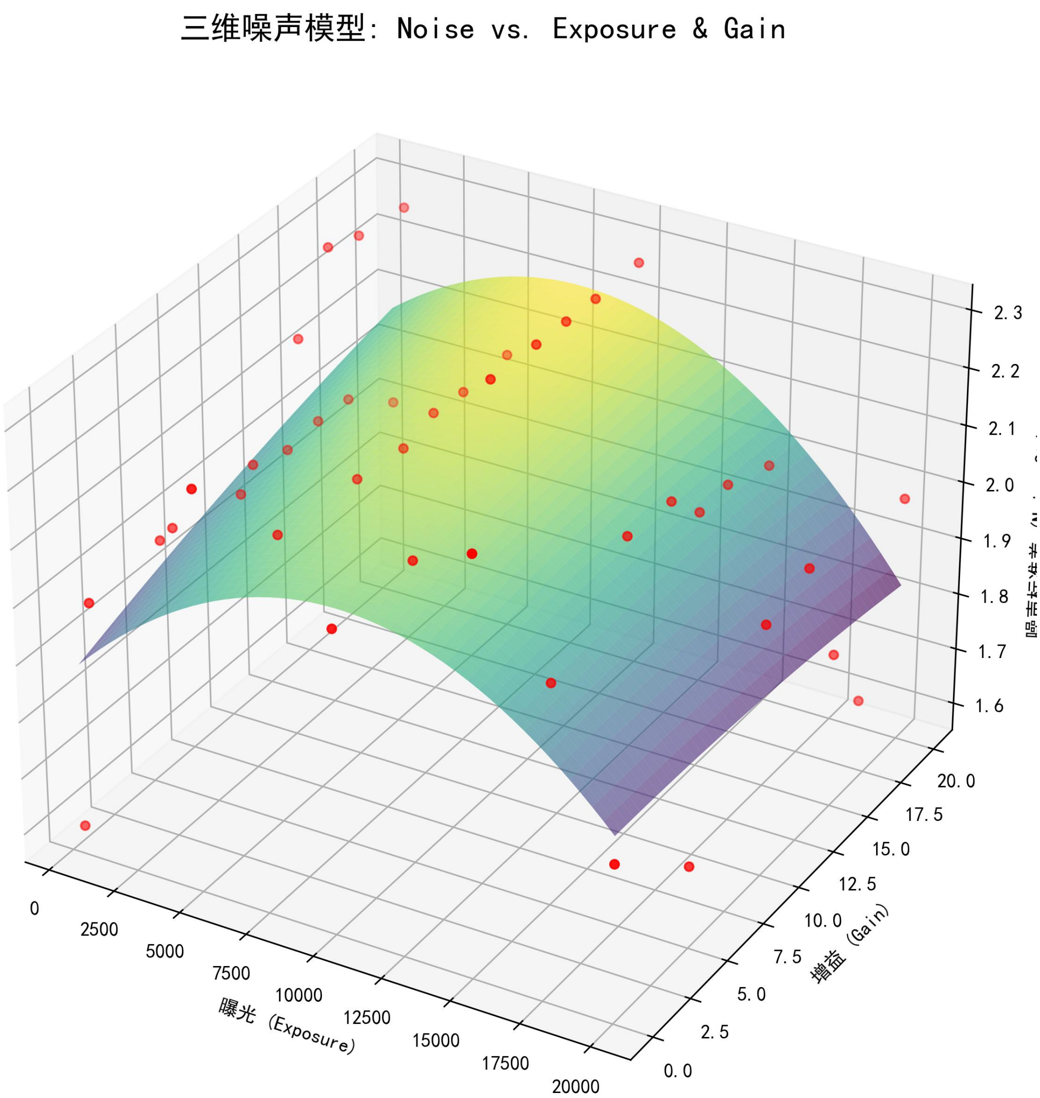
A joint model establishing noise as a function of both exposure time and gain was developed:

$$\text{noise} = f(\text{exp},\text{gain}) = p_{00} + p_{10}\,x + p_{01}\,y + p_{20}\,x^2 + p_{11}\,xy + p_{02}\,y^2$$

where x represents exposure time and y represents gain value.

```
def fit_and_plot_3d_model(result_root_dir, noise_data):
    # 1. 提取三维数据
    exposures = np.array([d['exp'] for d in noise_data])
    gains = np.array([d['gain'] for d in noise_data])
```

```python
noise_stds = np.array([d['noise_std'] for d in noise_data])

# 2. 定义二阶多项式三维模型
def poly_2d(X, p00, p10, p01, p20, p11, p02):
    x, y = X
        return p00 + p10*x + p01*y + p20*x**2 + p11*x*y + p02*y**2

    # 3. 拟合三维模型，计算整体拟合优度
    try:
        params, _ = curve_fit(poly_2d, (exposures, gains), noise_stds)
        y_fit = poly_2d((exposures, gains), *params)
        r2 = r2_score(noise_stds, y_fit)
```



$$f(x, y) = 1.8631 \times 10^{-8} + 5.0495 \times 10^{-13} x + 1.259 \times 10 - 10y - 2.4903 \times 10^{-17} x^2 - 6.0665 \times 10^{-15} xy - 1.4041 \times 10^{-12} y^2$$

$$R^2 = 0.4394$$

2.4 Conclusions and Discussion

This experiment investigated the relationship between noise standard deviation and two key imaging parameters, exposure time and gain, and constructed a quantitative noise model.

When the gain was fixed at 5.0, the quadratic model fitting for the relationship between noise standard deviation and exposure time achieved an excellent goodness of fit with an R^2 value of **0.9921**, indicating that the quadratic model can well describe the variation pattern of noise with exposure time under this condition.

For the relationship between noise standard deviation and gain with the exposure fixed at 5000 μs, among the linear model, square root model, and mixed model, **the mixed model emerged as the optimal one**. It obtained an R^2 value of **0.9930**, which is a very high goodness of fit. This shows that the mixed model, which combines the effects of both square-root and linear terms of gain, can excellently capture the complex relationship between noise and gain.

Regarding the three-dimensional noise model established by simultaneously considering exposure time and gain, it aims to comprehensively describe the combined effect of these two parameters on noise. However, the model only achieved an R^2 value of 0.4394. This relatively low R^2 value suggests that although the model can capture some of the combined effects, there are still other factors or more complex relationships between exposure time, gain, and noise that are not fully accounted for in the current three-dimensional model. Future work could focus on refining the three-dimensional model by incorporating additional influencing factors or exploring more complex model structures to better describe the joint effect of exposure time and gain on noise.