# 数学公式检测


> 北京航空航天大学
>
> 张杰宣


## 简单使用

打开 predict.py 即可使用，其中 predict.py 的 mode 有：

- predict 单张图片预测，可以直接使用，无需更改任何内容
- dir_predict 对一个文件夹下所有文件预测，并输出预测图片到指定文件夹，在使用时，请注意更改文件夹的 Path 路径：

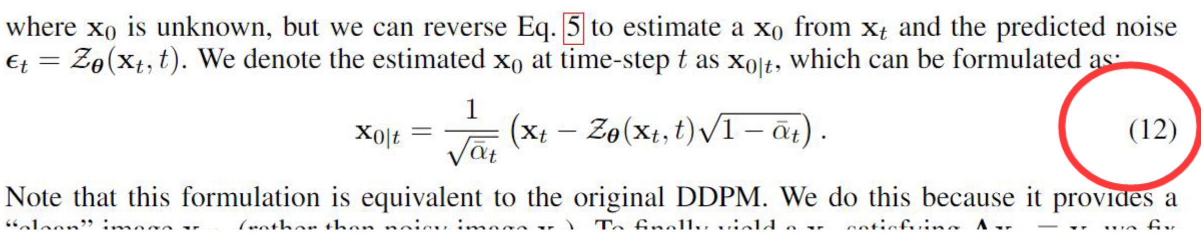
  ```
  dir_origin_path = 'pdf_img'
  dir_save_path = "pdf_img_out"
  ```

## 类别以及框的颜色

我们在本次检测任务中设置了两个类别：

- display 对应红色框，只包含公式本身，不包含序号
- display_all 对应蓝色框，包含公式以及对应的序号

具体的区别就是是否含有编号，如下图：

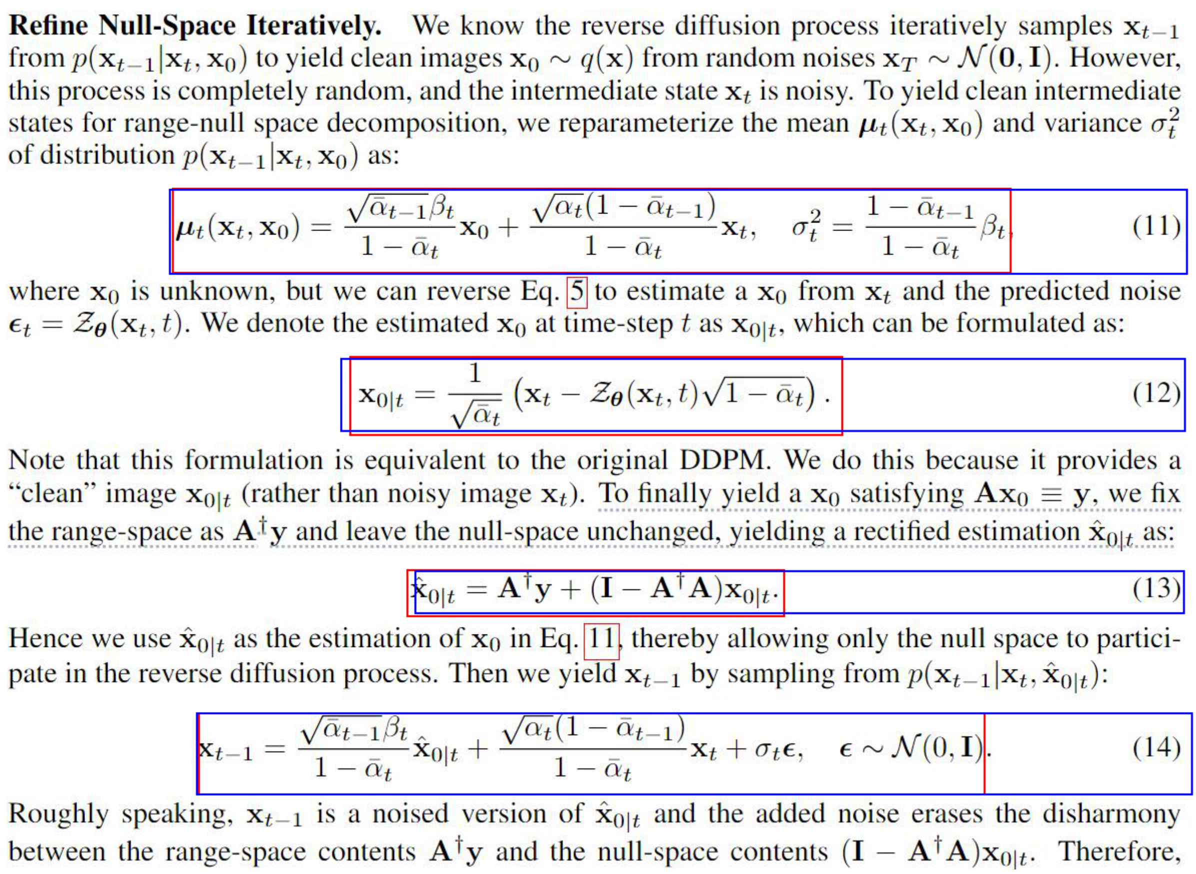


where $\mathbf{x}_0$ is unknown, but we can reverse Eq. 5 to estimate a $\mathbf{x}_0$ from $\mathbf{x}_t$ and the predicted noise $\epsilon_t = \mathcal{Z}_{\boldsymbol{\theta}}(\mathbf{x}_t, t)$. We denote the estimated $\mathbf{x}_0$ at time-step $t$ as $\mathbf{x}_{0|t}$, which can be formulated as:

$$\mathbf{x}_{0|t} = \frac{1}{\sqrt{\bar{\alpha}_t}}\left(\mathbf{x}_t - \mathcal{Z}_{\boldsymbol{\theta}}(\mathbf{x}_t, t)\sqrt{1-\bar{\alpha}_t}\right). \quad (12)$$

Note that this formulation is equivalent to the original DDPM. We do this because it provides a

## 测试效果



**Refine Null-Space Iteratively.** We know the reverse diffusion process iteratively samples $\mathbf{x}_{t-1}$ from $p(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)$ to yield clean images $\mathbf{x}_0 \sim q(\mathbf{x})$ from random noises $\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. However, this process is completely random, and the intermediate state $\mathbf{x}_t$ is noisy. To yield clean intermediate states for range-null space decomposition, we reparameterize the mean $\boldsymbol{\mu}_t(\mathbf{x}_t, \mathbf{x}_0)$ and variance $\sigma_t^2$ of distribution $p(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)$ as:

$$\boldsymbol{\mu}_t(\mathbf{x}_t, \mathbf{x}_0) = \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}\mathbf{x}_0 + \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}\mathbf{x}_t, \quad \sigma_t^2 = \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t \tag{11}$$

where $\mathbf{x}_0$ is unknown, but we can reverse Eq. 5 to estimate a $\mathbf{x}_0$ from $\mathbf{x}_t$ and the predicted noise $\boldsymbol{\epsilon}_t = \mathcal{Z}_{\boldsymbol{\theta}}(\mathbf{x}_t, t)$. We denote the estimated $\mathbf{x}_0$ at time-step $t$ as $\mathbf{x}_{0|t}$, which can be formulated as:

$$\mathbf{x}_{0|t} = \frac{1}{\sqrt{\bar{\alpha}_t}}\left(\mathbf{x}_t - \mathcal{Z}_{\boldsymbol{\theta}}(\mathbf{x}_t, t)\sqrt{1-\bar{\alpha}_t}\right). \tag{12}$$

Note that this formulation is equivalent to the original DDPM. We do this because it provides a "clean" image $\mathbf{x}_{0|t}$ (rather than noisy image $\mathbf{x}_t$). To finally yield a $\mathbf{x}_0$ satisfying $\mathbf{A}\mathbf{x}_0 \equiv \mathbf{y}$, we fix the range-space as $\mathbf{A}^{\dagger}\mathbf{y}$ and leave the null-space unchanged, yielding a rectified estimation $\hat{\mathbf{x}}_{0|t}$ as:

$$\hat{\mathbf{x}}_{0|t} = \mathbf{A}^{\dagger}\mathbf{y} + (\mathbf{I} - \mathbf{A}^{\dagger}\mathbf{A})\mathbf{x}_{0|t}. \tag{13}$$

Hence we use $\hat{\mathbf{x}}_{0|t}$ as the estimation of $\mathbf{x}_0$ in Eq. 11, thereby allowing only the null space to participate in the reverse diffusion process. Then we yield $\mathbf{x}_{t-1}$ by sampling from $p(\mathbf{x}_{t-1}|\mathbf{x}_t, \hat{\mathbf{x}}_{0|t})$:

$$\mathbf{x}_{t-1} = \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}\hat{\mathbf{x}}_{0|t} + \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}\mathbf{x}_t + \sigma_t\boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I}). \tag{14}$$

Roughly speaking, $\mathbf{x}_{t-1}$ is a noised version of $\hat{\mathbf{x}}_{0|t}$ and the added noise erases the disharmony between the range-space contents $\mathbf{A}^{\dagger}\mathbf{y}$ and the null-space contents $(\mathbf{I} - \mathbf{A}^{\dagger}\mathbf{A})\mathbf{x}_{0|t}$. Therefore,

我们在默认文件夹下已经放置了几张图片，并进行了处理，想观察直接效果可以观看 pdf_img_out 文件夹内容。

## 如何拿到预测框的数值?

我们在 Inference 的过程中，同时将预测框也进行了输出，如代码中的:

```
pre_image, bboxes = yolo.detect_image(image)
```

bboxes 存储着该张图片的所有预测框信息，格式为 list[(x_min, y_min, x_max, y_max)]，其中 list 的长度为该张图片拥有多少个预测框。