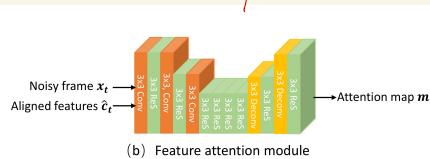
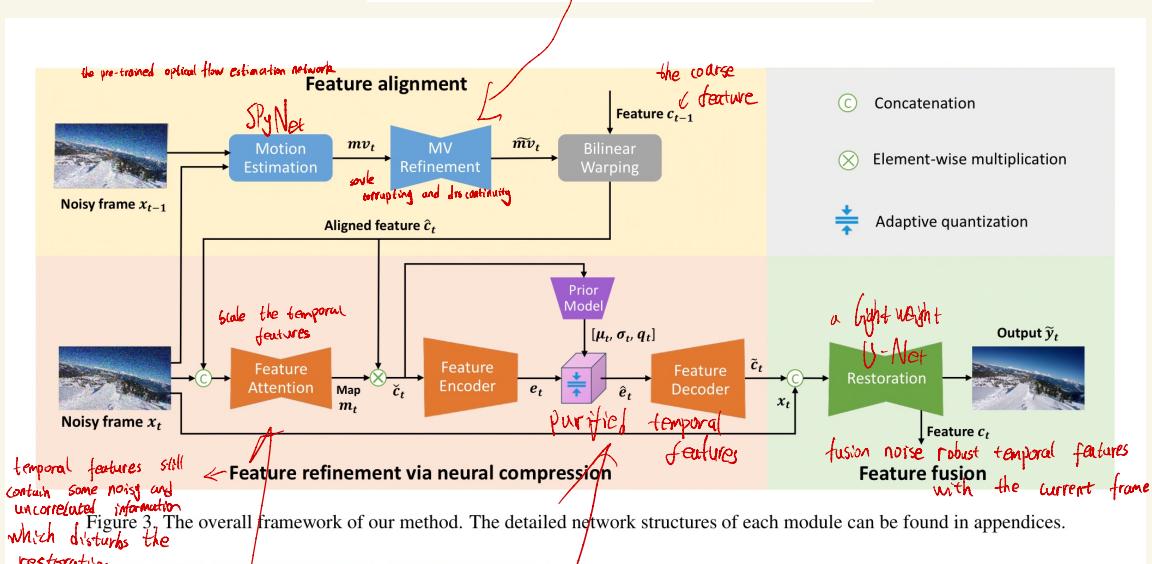


(a) MV refinement module



(b) Feature attention module

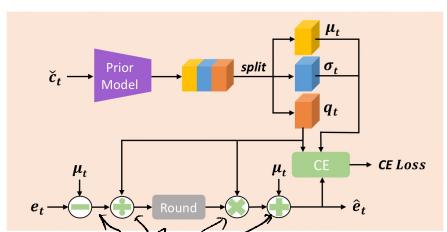


Figure 5. The illustration of adaptive quantization mechanism. CE means cross-entropy.

这是本文的创新点，
即先降噪，后量化。但对不同的帧使用不同的量化步长，引入自适应。
 μ 是均值， σ 是方差，方差越大可以反向图像与噪声的关系，通过自己去猜测噪音和以便能够让图像对不同区域进行去噪。

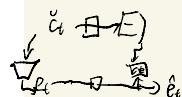
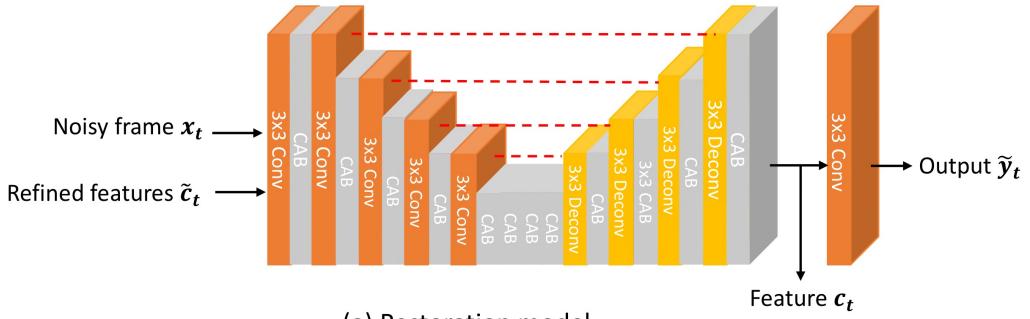
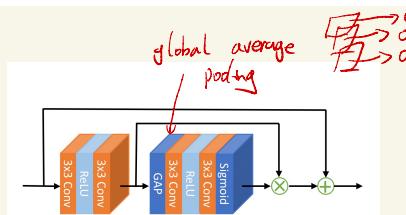


Figure 9. The structure of neural compression-based feature learning module.



(a) Restoration model



(b) CAB block
channel - attention block
extract the features at each scale

SPyNet: Optical Flow Estimation using a Spatial Pyramid Network

光流估计
金字塔

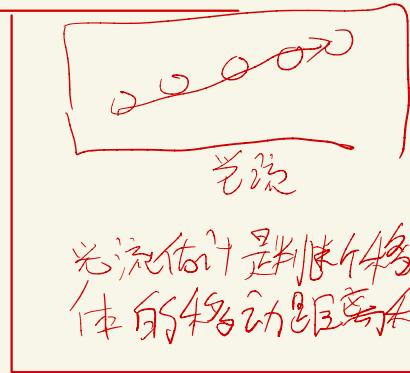
Problem

用卷积网络体系结构,当帧之间的运动大于一个(或几个)像素时,时空卷积器将无法获得有意义的响应。换而言之,如果一幅图像中的卷积窗口到处没有与相关图像像素重叠,则无法学习有意义的时空卷积器。
 \rightarrow detail, sub-pixel, optical flow and precise motion boundaries
 (用深度学习)
 long-range correlations (用现有方法)

$d(\cdot)$ downsampling function

$$d(I) = I' \quad d(\cdot) = u(\cdot)$$

$$\begin{matrix} m \times n \\ \downarrow & \downarrow \\ \frac{m}{2} \times \frac{n}{2} \end{matrix}$$



V the optical flow field

$w(I, V)$ according to V

\uparrow
Warp
function
 $w(I, V)$

$\{G_0, \dots, G_K\}$
a set of trained
 CNN model

$$\hat{V}_k = \hat{V}_{k-1} + u(\hat{V}_{k-1})$$

$$\begin{matrix} \uparrow & \uparrow \\ \text{target} & \text{target} \\ \text{residual} & \text{flow} \\ \text{flow} & \end{matrix}$$

$$\hat{V}_k = G_k(I_k, w(I_{k-1}, u(V_{k-1})))$$

residual flow

\hat{V}_k \uparrow
 I_k \uparrow
 pyramid level

$$\hat{V}_k = u(\hat{V}_{k-1}) + \hat{V}_k$$

$$\hat{V}_0 = \hat{V}_0 \quad \{I_0^1, I_0^2\}$$

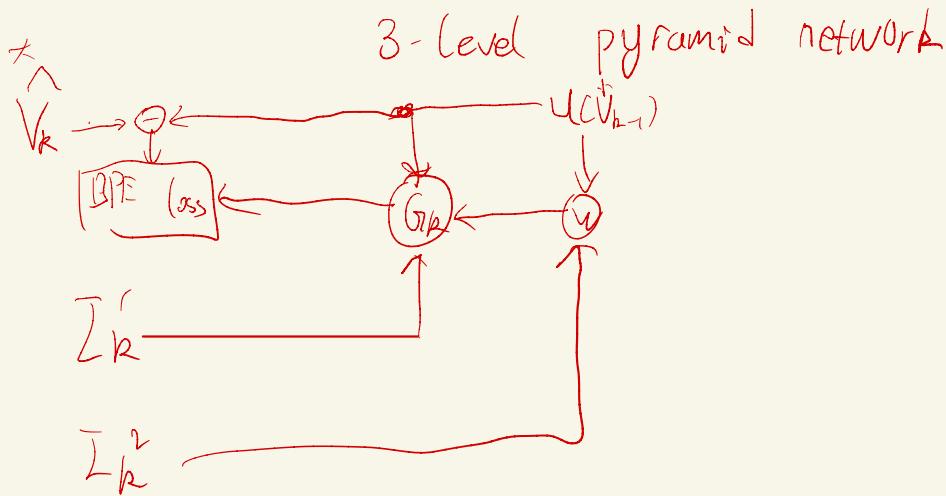
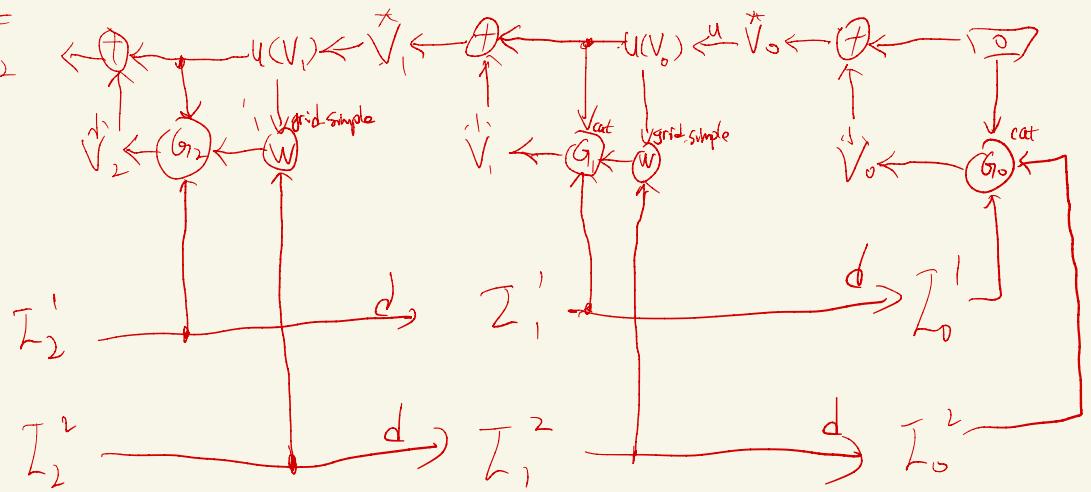


Image warping: 在任何图像中所描绘的任何形状
都产生显著的拉伸扭曲可用于校正图像偏移。

$$y_i = \frac{x_i}{(\beta_i^2 + \sum r_i x_i^2)^{\frac{1}{2}}}$$

$$\beta_{\min} = 10^{-6}$$

$$\beta_{\text{bound}} = [\beta_{\min} + b^2]^{\frac{1}{2}}$$

$$r_{\text{init}} = 0.1$$

$$r_{\text{bound}} = b$$

$$b = 2^{-18}$$

Param

$$\beta = (\underbrace{[1, 1, \dots, 1]}_{ch} + b^2)^{\frac{1}{2}}$$

$$r = (r_{\text{init}} \times \underbrace{\begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 \end{bmatrix}}_{ch \times ch} + b^2)^{\frac{1}{2}}$$

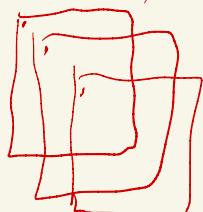
$$\beta = \beta^2 - b^2$$

$$\beta = \max(\beta, \underbrace{[\dots]}_{cha} \times \beta_{\text{bound}})$$

$$r = r^2 - b^2$$

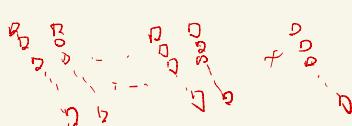
$$r = \max(r, \underbrace{[\dots]}_{cha \times ch} \times r_{\text{bound}})$$

$$(18, 32, 32)$$

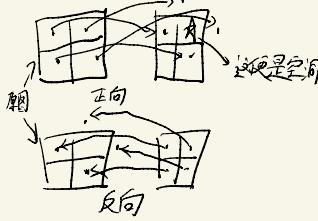


$$(8, 8, 1, 1)$$

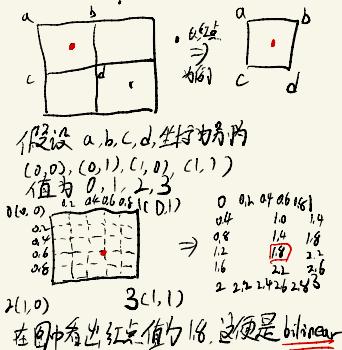
$$(8)$$



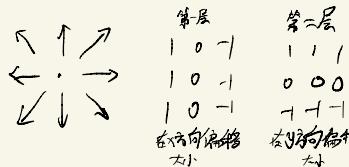
逆向的 Bilinear Warping
是 backward image warping
又称为 inverse warping
正向 warping 没有反向 warping 是因为
是因为正向 warping 会产生大量空洞导致图像产生噪音



反向的原图也有空洞,说明我们
使用 bilinear 操作



在前面看出是从原图中取出某些值
在 spy net 中, 原因是原汁原味的三通道(RGB)的图片, spy net 是光流
估计模型, 我们会提取图像中各种
主体的运动方向, 用一个两层向量表示 J2mv



spy net 是用 mv 去提取第二张图(前一帧与该帧,
此时是逐帧图像) \uparrow warp

下面看如何提取的, 如何 warp?

首先 初始化一个简单且与原图同等大小的
MV, 简单的含义是利用此时的 MV 去
直接 warp 原图会得到与原图一模一
样的结果。

然后将此时的 MV 加上我们输入
的 mv, 没错我将运动的 mv 加进来了,
此时 MV + mv 再去原图中 warp 出来
的 mv 更加的清晰, 因为 spy net 是从两
张有运动差别的图中学到准确的 mv,
它输入 warp 的 mv 是降级前的 mv, 所以
spy net 是从不同的清晰程度学到准确的 mv,
而 Feature alignment 的 warp 是不断的
将前一帧 C 和此帧的 mv 进行 warp, 也就
是利用此帧的 mv 去提取前一帧的 C,
这些做的目的不妨猜测一下, 在雾雨等干
扰中, 精准的提取运动目标。

之前的不连贯性和毁坏很像是对
运动物体而言的。

对齐我感觉就像马老师之前说过
那阵子, 一个物体在两个时间点定位



即在两个帧下物体的
对齐