机器学习时代的经典视频去噪方法：兼具鲁棒性、高效性与可控性

Xin Jin[1*] Simon Niklaus[2†] Zhoutong Zhang[3] Zhihao Xia[3]
Chunle Guo[1,4‡] Yuting Yang[3] Jiawen Chen[3] Chongyi Li[1,4]
[1]VCIP, CS, Nankai University [2]Adobe Research [3]Adobe [4]NKIARI, Shenzhen Futian
xjin@mail.nankai.edu.cn, sniklaus@adobe.com,
{guochunle,lichongyi}@nankai.edu.cn,
https://srameo.github.io/projects/levd

Figure 1. 在真实场景的视频中，噪声的表现形式可能存在极大差异；当输入数据严重偏离训练分布时，有监督方法便会失效。视频素材由 Robert Kjettrup 提供，人脸区域在推理完成后进行了像素化处理。我们为重新训练的模型标注符号 †。

Abstract

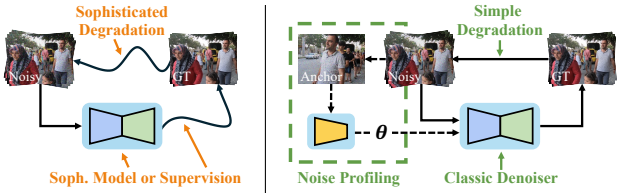
在众多视频处理任务中，去噪是一项至关重要的步骤，尤其在交互式编辑等应用中，对图像质量、处理速度以及用户可控性提出了极高要求。近年来，借助深度学习的去噪方法在效果上取得了显著提升，但由于训练数据分布与真实视频中复杂多变的噪声模式之间存在差异，这些方法在实际应用中常常出现不可预期的失效情况，同时在运行效率和用户可控性方面也存在明显不足。相较之下，传统的去噪方法在自然场景视频中表现更加稳健，并能在现代硬件上高效运行，但其参数需针对每段视频手动调节，过程既繁琐又依赖经验。为弥合这两类方法的差距，本文提出一种基于传统方法的可微分去噪流程，并进一步引入神经网络，根据每个具体输入预测最优去噪参数，从而实现了一种兼具鲁棒性、高效性与可控性的视频去噪方法。

1. 简介

视频去噪是任何视频编辑流程中的基础环节，通常在进行调色之前应用。通过对专业视频编辑人员的访谈，我们发现他们不仅追求干净的去噪结果，还希望去噪过程快速高效，以避免中断工作流，并希望具备一定的可控性，以体现其艺术表达。具体而言，编辑人员常常需要在“保留一定噪声以避免过度平滑”与“彻底去噪”之间做出权衡，而不同场景下的选择也可能截然不同。正因如此，像 Arri Alexa 这样的专业摄像机甚至允许用户选择不同的噪声模型，以更好地适应后期制作中的多样化工作流。

然而，近期的视频去噪研究往往仅关注第一个方面——去噪质量，这使得这些方法难以适用于典型的视频编辑工作流。同时，我们也发现，即便这些方法专注于提升质量，依然可能在某些场景中出现失败，如 Fig. 1 所示的例子。这其实并不令人意外，因为真实视频中的噪声表现形式极为多样。除了图像常见的退化方式外，视频还受到时间压缩的影响。例如，在 H.264 编码中，P 帧与 B 帧会从 I 帧中复制信息，这意味着噪声常常在时间上具有相关性。此外，不同的视频编码器也会以不同方式影响噪声分布，即使使用相同的编码标准，采用不同的编码器或设置也可能导

*This project was done during Xin Jin's internship at Adobe Research.
†Project lead. ‡ Corresponding author.



(a) Related Work (b) Ours

Figure 2. 现有相关方法（左）与我们提出方法（右）在视频去噪任务中的高层次对比示意图。

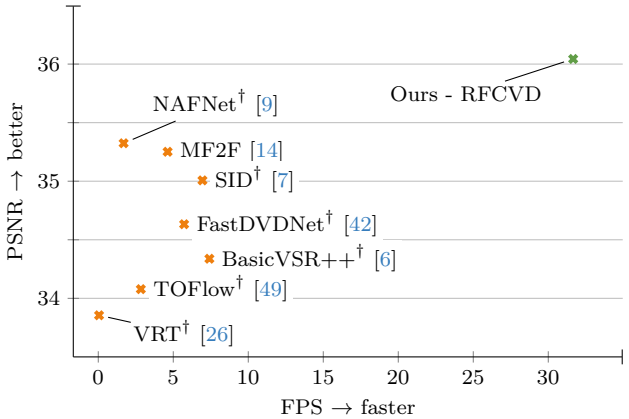


Figure 3. 在 CRVD（sRGB）基准测试集 [50] 上进行的视频去噪性能评估。图中以 PSNR 衡量所有 ISO 设置下的去噪质量，并以 FPS（每秒帧数）衡量在 RTX 3090 GPU 上的计算效率。我们对部分模型进行了重新训练以提升性能，使用 † 符号标注。

致噪声特性发生显著变化。

为了在深度学习背景下应对视频噪声的复杂性，正如 Fig. 2(a) 所示，已有方法主要从以下几个方向展开：提出更复杂的监督方式以训练模型 [14, 40]，设计精细的噪声模拟流程 [46]，或基于天然不易过拟合的图像生成建模方式 [13]，或将以上方法进行组合。然而，正如 Fig. 1 中所展示的例子，这些策略在某些情境下仍难以满足实际需求。

相比之下，传统去噪方法不仅表现良好，而且在现代硬件上运行速度较快。然而，这类方法需要手动调节参数，这不仅繁琐，而且对操作者的技能有一定要求。我们认为这是一个机遇。具体而言，如 Fig. 2 (b) 所示，通过以可微分的方式实现传统去噪流水线，我们可以让神经网络学习预测针对给定输入的最优去噪参数。正如 Fig. 3所展示的，这种方法不仅鲁棒性强，同时速度较快，并且支持用户对去噪过程进行控制。需要明确的是，若已知噪声分布特征，例如为特定摄像机模型开发针对受限视频编码器的去噪器时，任何合理的深度学习方法均能提供更优的去噪效果，但此类方法在鲁棒性和可控性方面存在局限。

简而言之，我们的贡献包括：（1）提出一种在深度学习框架下利用传统视频去噪流水线的方法，通过将噪声分析与去噪过程解耦，避免重复计算以提升效率；（2）设计了一种不仅鲁棒且快速，同时具备良好可控性的实时视频去噪器（RFCVD）；（3）构建了一套基于加性白噪声与 H.264 转码相结合的训练增强流水线，且实验表现出乎意料的优异。

2. 相关工作

传统视频去噪。大多数传统视频去噪方法可被视为一系列线性与非线性滤波过程的组合，其中滤波强度通常是噪声水平估计的函数。这类方法中，许多基于块匹配技术 [10–12, 31]，通过在帧内及帧间聚合相似的图像块，再对其进行滤波和融合，以生成去噪结果。该流程有诸多变体，例如改进匹配过程 [28]，或结合双边滤波 [8, 18, 36, 43] 以实现时域与空域的联合去噪 [1, 37]。此外，为提高计算效率，可引入图像金字塔结构 [3, 16]。近年来，经典多帧去噪方法的一些变体甚至可以高效运行在普通智能手机上 [20, 27, 48]。

机器学习在视频去噪中的应用。近年来，借助机器学习技术处理视频去噪问题的研究日益增多。这些方法包括模仿传统去噪器执行显式匹配的策略 [13, 44, 49]，也包括仅隐式建模帧间关联的方法 [24, 32, 42]。虽然多数方法对每一帧进行独立去噪，但也存在类似无限脉冲响应（IIR）风格的方法，它们在处理过程中传递信息，使得已去噪帧可指导后续帧的恢复 [5, 6]。在网络架构方面，视觉与语言领域注意力机制的发展亦推动了多种基于注意力的视频复原方法的出现 [4, 26, 29, 41, 45]。然而，机器学习方法也面临一个核心挑战：一旦输入中出现训练阶段未见过的噪声模式，其去噪效果就会变得不可预测。

噪声模拟。为了提升方法对真实视频的泛化能力，常见做法是在合成噪声时采用更复杂的退化流程。例如，有方法提出进行两轮退化处理 [46]，或将退化过程进行随机重排 [51]，亦或引入更高级的操作符如色调映射 [52]。尽管这些策略在一定程度上增强了对真实视频噪声的鲁棒性，但它们只能缩小而无法完全弥合泛化性能的差距。

自监督学习。提升视频去噪泛化能力的另一策略是采用无需真实标签的自监督学习方式，直接在真实视频上进行训练。此类方法可借助回归损失本身的性质实现 [25]，或基于盲点网络的设计思想 [14, 17, 23, 40]。虽然这些方法使得在真实噪声视频上进行训练成为可能，但现实中很难收集包含所有类型噪声的视频数据集。因此，自监督学习往往也依赖于测试时自适应，而这本身也带来新的挑战。

3. 方法

视频去噪的基本步骤是首先分析噪声，然后再将其去除。相比之下，近年来的一些方法将这两个步骤合并为一个过程，通过一个神经网络同时完成噪声分析与去除：该网络以噪声图像序列作为输入，直接输出去噪结果。然而，在实际视频中，噪声特性通常在时间维度上是相对稳定的，而这些方法却对每一帧单独去噪，因而需要反复地对噪声进行分析，造成了冗余。

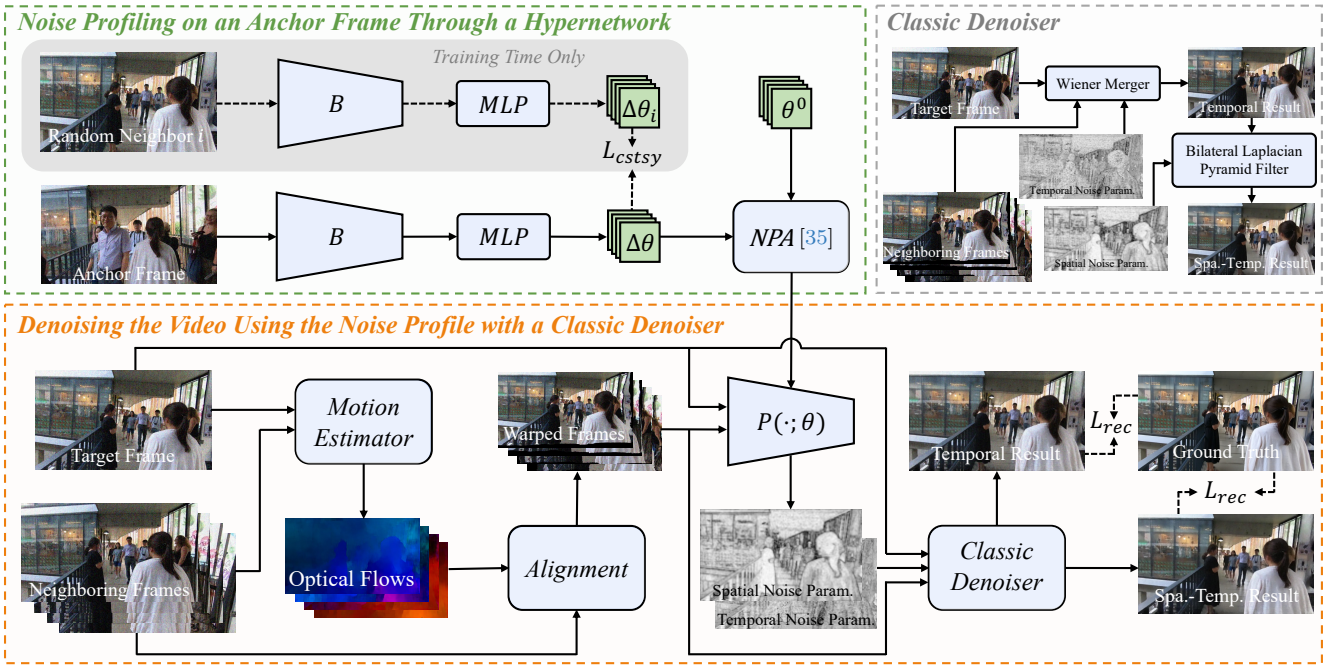


Figure 4. 视频去噪的基本流程包括两个阶段：首先分析噪声，随后执行去噪操作。我们的方法模仿了这一流程，先在一个随机选取的锚帧上估计噪声分布（图中左上角绿色部分），然后利用该噪声分布对整个视频进行去噪（图中下方黄色部分）。具体而言，我们采用了超网络结构，其中噪声分布 θ 实质上作为后续去噪器的参数。该去噪器为传统处理流程，包括：（1）使用维纳滤波器对通过光流对齐后的邻近帧进行时域去噪；（2）通过双边拉普拉斯金字塔滤波器对时域融合结果进行空域去噪。在此过程中，一个小型神经网络 $\mathcal{P}(\cdot;\theta)$ 用于预测维纳融合器与双边滤波器的空间变化参数。该任务分离策略提高了整体计算效率，避免了重复进行噪声分析的计算开销。 [35]

为避免这种冗余，我们遵循“先分析噪声，再进行去噪”的自然分工。如 Fig. 4所示，我们首先在一帧锚点图像上估计出噪声特征（即噪声画像），然后利用该特征对整段视频进行去噪。具体而言，我们采用了一个超网络结构，其中噪声画像 θ 实质上提供了后续去噪器的参数。我们的去噪器是一个传统的处理流程，包含两部分：（1）维纳滤波器，利用光流对齐的相邻帧进行时间域去噪；（2）双边拉普拉斯金字塔滤波器，对经时间融合后的图像进行空间域去噪。小型神经网络 $\mathcal{P}(\cdot;\theta)$ 用于预测 Wiener 融合模块及双边滤波器的空间自适应参数。

我们将依次讨论这些部分，之后再详细介绍用户控制和训练相关内容。

3.1. 噪声剖析

由于噪声通常并不是空间上均匀分布的，而是常常依赖于图像的信号强度，我们认为噪声分布不应仅仅是一组对整个输入图像统一适用的固定数值，而应是一种能够推导出空间可变去噪参数的描述子。我们认为这是超网络 [19] 的理想应用场景，其中噪声分析器用于估计一个小型神经网络 $\mathcal{P}(\cdot;\theta)$ 的权重 θ，该网络负责预测空间变化的去噪参数。然而，我们并不直接预测 θ，而是通过 NPA 方法 [35] 来稳定训练过程，其中 $\theta = \theta^0 + \Delta\theta$，$\theta^0$ 为可学习参数，$\Delta\theta$ 为由超网络生成的预测部分。

直观而言，分析一张图像的噪声分布是一个融合了低层图像处理与高层语义理解的过程。具体来说，模型需要首先理解图像内容，才能进一步检查细节，以区分非预期噪声与实际的纹理结构。因此，我们在超网络中使用了一个预训练的 ConvNext [30] 主干网络 $\mathcal{B}$，并连接一个随机初始化的 MLP 头部。我们发现这种主干网络对预测出高质量噪声分布至关重要。

最后，对于给定视频，我们应选择哪一帧作为锚帧来进行噪声分析？我们的目标是使这个选择对最终的去噪结果没有影响，即去噪性能应当不依赖于是否选择了“优质”的锚帧。为此，我们始终选择视频的第一帧作为锚帧，因为它与其他帧在本质上没有差异，并进一步引入一致性损失

$$\mathcal{L}_{cstsy} = \|\Delta\theta - \Delta\theta_i\|_2, \tag{1}$$

其中，我们鼓励超网络对锚帧 $\Delta\theta$ 的预测，与任意随机邻近帧 $\Delta\theta_i$ 的预测保持一致。

3.2. 视频去噪

一旦获得噪声分布参数 θ，我们逐帧处理输入视频的每个输出帧，通过一个包含三层卷积的小型神经网络 $\mathcal{P}(\cdot;\theta)$ 获取空间可变的去噪参数。随后，我们利用这些参数依次进行时间去噪和空间去噪。

在时间去噪阶段，为去噪目标帧，首先对其前后各两帧进行对齐。具体而言，我们采用现成的 SpyNet [38] 以目标分辨率的四分之一进行光流估计，以提升计算效率及对噪声的鲁棒性，并利用估计的

Figure 5. 通过缩放由 $\mathcal{P}(\cdot;\theta)$ 估计的去噪参数，我们的框架能够方便地调节时间去噪强度（通过 σ^2）以及空间去噪强度（通过 σ_d 和 σ_r）。在本示例中，对亮度和色度通道采用了统一的参数缩放，但它们也可以分别调整以实现更细致的控制。详见补充材料中的视频演示，展示用户如何以交互方式控制去噪效果。视频素材由 Amit Zinman 提供。

光流将邻近帧变形对齐至目标帧。接着，按照 Hasinoff et al. [20] 的方法，使用维纳滤波器合并对齐后的帧，但滤波块大小设为 8×8，且使用来自 $\mathcal{P}(\cdot;\theta)$ 的噪声图代替每块噪声的均方根近似。为便于用户控制，亮度和色度通道的维纳滤波独立执行。

随后，我们通过三层双边拉普拉斯金字塔滤波对时间去噪后的目标帧进行空间去噪。该方法借鉴了多分辨率双边滤波 [53] 的思想，但采用拉普拉斯金字塔替代高斯金字塔，并在首层金字塔中使用全程双边滤波替代小波阈值处理。具体来说，我们让 $\mathcal{P}(\cdot;\theta)$ 估计噪声图金字塔，并将其作为双边滤波的参数 σ_s 和 σ_r。此外，与 Wiener 滤波类似，亮度和色度通道的空间去噪同样独立执行，以增强用户的可控性。

3.3. 用户控制

通过对多位创意行业专业人士的访谈，我们发现他们在对视频片段进行去噪时希望能够表达自身的艺术风格。也就是说，他们常常需要在保留一定噪声与避免过度平滑之间做出权衡。我们方法中实现用户可控性的关键在于由 $\mathcal{P}(\cdot;\theta)$ 生成的空间变化参数，这些参数用于引导时域和空域的去噪过程。具体而言，在维纳滤波中，我们使用了两张用于 σ^2 的参数图（一张用于色度通道，一张用于亮度通道）；而在双边拉普拉斯金字塔滤波中，我们分别为 σ_d 和 σ_r 提供了两组金字塔结构的参数图（同样，一组用于色度，一组用于亮度）。总体而言，这构成了六组可供用户调控的参数。

那么，我们是如何对这些参数进行调节的呢？如 Fig. 5 所示，我们发现仅对这些空间变化的去噪参数进行统一缩放就能获得相当不错的效果，因此我们可以向最终用户提供六个可调节的控制滑块。由于我们的去噪框架支持实时运行，调整滑块后能够立即获得反馈，从而使用户更容易获得期望的视觉效果。

3.4. 增强流程

为了训练我们的模型，我们需要成对的干净与带噪帧序列，这与其他非自监督学习的去噪方法相同。具体而言，我们采用了一种典型范式：使用一组“无噪声”视频作为真实标签，并通过各种退化方式对其进行增强，从而获得相应的带噪输入。为了确保该方法效果良好，我们发现在退化流程中引入时间压缩是至关重要的，因为这会使噪声在时间维度上具有相关性。

我们首先使用 REDS 数据集 [33] 获取无噪声视频。该数据集包含 240 个视频，拍摄帧率为相对较高的 120 帧每秒，这使我们可以通过下采样来增强帧率多样性。我们从这些视频中随机提取帧序列，首先对其添加服从 $\mathcal{U}(1,50)$ 的方差采样的高斯白噪声，随后使用 H.264 编解码器通过 libx264 进行转码，CRF 值从 $\mathcal{U}(18,30)$ 区间随机采样，以使噪声在时间上具有相关性。

我们发现该数据处理流程效果出乎意料地好。实际上效果好到我们对比评测中的其他模型都进行了重新训练，因为使用我们重新训练的版本在 CRVD 基准测试 [50] 上获得了远超原始预训练权重的结果。然而，我们没有重新训练 Real-ESRGAN [46]，因为该方法自带更复杂的数据增强流程；同时我们也没有重新训练 MF2F [14] 和 UDVD [40]，因为它们采用了较为复杂的监督策略，我们难以在公平和完整的前提下加以复现。

3.5. 实现细节

关于架构细节，我们在超网络中采用了 ConvNext [30] 的基础版本作为骨干网络 $\mathcal{B}$。为了将该骨干网络的输出映射为残差噪声特征 $\Delta\theta$，我们使用了一个包含五层的 MLP，并在各层之间使用了 PReLU 激活函数 [21]。对于预测空间可变去噪参数的神经网络 $\mathcal{P}(\cdot;\theta)$，我们采用了三层卷积层，步长为 2，层间同样使用 PReLU 激活。为了保证输出的去噪参数非负，我们对该网络的输出使用了 softplus 函数约束 [15]。此外，我们发现除了将（对齐后的）帧输入 $\mathcal{P}(\cdot;\theta)$ 外，同时输入由 Sobel 滤波器近似计算的梯度信息及指示对齐帧中像素有效性的掩码（用于标识像素是否由帧外区域变换而来）[2]，能进一步提升效果。

在重建损失中，我们最小化时序去噪图像 I^{T} 与真实图像 I^{GT} 之间的差异，同时也最小化空间去噪金字塔各级 I^{ST} 与对应真实图像 I^{GT} 的差异，其形式为，

	PSNR delta (higher PSNR is better)		SSIM delta (higher SSIM is better)		LPIPS delta (lower LPIPS is better)	
NAFNet [9]	30.25	–	0.747	–	0.358	–
NAFNet† [9]	35.32	+ 5.07	0.937	+ 0.190	0.089	- 0.269
FastDVDNet [42]	27.89	–	0.565	–	0.502	–
FastDVDNet† [42]	34.63	+ 6.75	0.914	+ 0.349	0.124	- 0.378
TOFlow [49]	25.96	–	0.673	–	0.251	–
TOFlow† [49]	34.08	+ 8.11	0.903	+ 0.231	0.147	- 0.104
BasicVSR++ [6]	31.98	–	0.769	–	0.326	–
BasicVSR++† [6]	34.34	+ 2.35	0.873	+ 0.104	0.167	- 0.158
VRT [26]	31.99	–	0.784	–	0.296	–
VRT† [26]	33.86	+ 1.86	0.848	+ 0.064	0.192	- 0.104

Table 1. 在 CRVD（sRGB）基准上的去噪结果。对比了多个方法，包括其原始版本和我们重新训练的版本（用 † 表示）。

	PSNR delta (higher PSNR is better)		SSIM delta (higher SSIM is better)		LPIPS delta (lower LPIPS is better)	
Avg. w/ $\mathcal{L}_{cstsy}$	35.95	–	0.9477	–	0.0757	–
first frame ← Ours	36.04	+ 0.09	0.9472	- 0.0005	0.0763	+ 0.0006
middle frame	35.92	- 0.03	0.9480	+ 0.0003	0.0753	- 0.0003
last frame	35.90	- 0.06	0.9480	+ 0.0003	0.0754	- 0.0003
Avg. w/o $\mathcal{L}_{cstsy}$	35.86	–	0.9460	–	0.0754	–
first frame	35.80	- 0.05	0.9454	- 0.0005	0.0750	- 0.0004
middle frame	36.02	+ 0.16	0.9473	+ 0.0014	0.0753	- 0.0001
last frame	35.75	- 0.11	0.9451	- 0.0008	0.0759	+ 0.0005

Table 2. 在 CRVD 基准数据集上，不同锚帧选择下的去噪结果。我们发现所提出方法对锚帧的选择具有较强的鲁棒性（上），这一点在一定程度上得益于 $\mathcal{L}_{cstsy}$ 的引入（下）。

对于 I^{ST} 与 I^{GT} ：

$$\mathcal{L}_{rec} = \left\| I^{\mathrm{T}} - I^{\mathrm{GT}} \right\|_2 + \sum_{l=1}^{3} \left\| I_l^{\mathrm{ST}} - I_l^{\mathrm{GT}} \right\|_2, \quad (2)$$

其中 l 表示空间去噪金字塔的第 l 级。需要注意的是，我们还在噪声剖析部分定义了保持一致性的损失，如式 (1)所示。

训练时，我们采用 Adam 优化器 [22]，初始学习率为 2×10^{-4}，并通过余弦退火调度逐步衰减至 1×10^{-7}。总训练迭代次数为 40 万次，批量大小为 24，训练样本为 512×512 像素的图像块。由于我们的管线计算负担较轻，训练时间不到两天。

4. 实验

我们在两类视频上对所提出的方法进行了定量评估：真实视频与合成视频。对于前者，我们采用了 CRVD 基准数据集 [50]，该数据集包含 RAW 格式的视频，我们通过提供的深度 ISP 模型 [39] 将其转换为 sRGB 格式。对于后者，我们使用 REDS 数据集 [33] 中的验证样本，并对其进行了两种方式的数据增强，这两种方式与训练时所采用的加性高斯白噪声（AWGN）与 H.264 转码的退化策略上具有显著差异。具体而言，其中一种方式是加入胶片颗粒噪声 [34] 并采用 AV1 转码，另一种方式则是加入具有空间相关性的高斯噪声，并使用 H.265 进行转码。在评估指标方面，我们遵循常规范式，展示 PSNR、SSIM [47] 和 LPIPS 指标 [54]。鉴于我们主张本方法具有较高的计算效率，我们还展示了帧率（FPS），该指标是在配有 CUDA 同步机制的 RTX 3090 GPU 上测得的。

在定性评估的数据方面，我们联系了业内的专业人士，他们向我们提供了一些在去噪过程中遇到困难的视频素材。由于这些专业人员没有获得所有演员的授权同意书，我们有时需要在推理后对面部进行匿名处理。

最后，我们将所提出的方法与多种类型的去噪器进行了比较，包括图像去噪方法 [7, 9]、视频去噪方法 [42, 49]、循环结构方法 [6]、Transformer 架构 [26]、采用复杂增强流程的方法 [46]，以及利用自监督机制的方法 [14, 40]。鉴于我们发现所提出的数据增强流程效果显著，我们对所有既未使用复杂增强流程、也未采用自监督机制的方法进行了重新训练。这样可以实现更为公平的比较，因为如 Tab. 1所示，我们通过重新训练版本在 CRVD 基准上提升了它们的性能。在本文中，为避免任何潜在的混淆，我们用 † 标注所有重新训练过的模型。

4.1. 定量对比

请参考 Tab. 3，该表总结了我们在 CRVD 基准上的定量评估结果。简而言之，我们的方法不仅在整体性能上表现最优，而且在速度方面也比第二快的方法快了四倍。我们将这一优异表现归因于对经典去噪技术的充分利用，这在很大程度上提升了模型对未知噪声模式的泛化能力。具体而言，我们的方法仅依赖少量参数驱动去噪过程，从而天然地减少了过拟合的风险，有助于缩小领域差距。为了验证这一假设，我们设计了一个基于合成数据的实验，其中所有模型均在带有 H.264 转码的 AWGN 数据上进行训练，但测试时使用了两种不同的退化策略。如 Tab. 4 所示，在该评估中我们的方法依然表现优越，从而进一步验证了我们最初的假设。

4.2. 定性比较

为了在真实环境中测试视频去噪性能，我们联系了多位创意行业的专业人士，他们慷慨地提供了在实际拍摄中难以去除噪声的素材。我们在 Fig. 6中展示了这些素材上去噪结果的代表性片段，完整的各方法视频结果请参阅补充材料。与定量评估中的结论类似，我们发现我们的方法在该真实测试中也表现良好。我们再次推测，这种优异表现归因于我们方法中对经典去噪技术的有效利用。

	ISO 1600		ISO 3200		ISO 6400		ISO 12800		ISO 25600		Overall		Speed	
	PSNR (higher PSNR is better)	rank	PSNR (higher PSNR is better)	rank	PSNR (higher PSNR is better)	rank	PSNR (higher PSNR is better)	rank	PSNR (higher PSNR is better)	rank	PSNR (higher PSNR is better)	rank	FPS (higher FPS is better)	rank
SID† [7]	38.85	7th of 10	37.68	7th of 10	35.82	4th of 10	33.51	4th of 10	29.18	3rd of 10	35.01	4th of 10	6.95	3rd of 10
NAFNet† [9]	39.48	3rd of 10	38.12	5th of 10	35.94	3rd of 10	33.53	3rd of 10	29.55	2nd of 10	35.32	2nd of 10	1.69	7th of 10
FastDVDNet† [42]	39.16	5th of 10	37.92	6th of 10	35.60	5th of 10	32.56	5th of 10	27.93	6th of 10	34.63	5th of 10	5.72	4th of 10
TOFlow† [49]	38.25	8th of 10	36.97	8th of 10	34.90	7th of 10	32.21	6th of 10	28.07	5th of 10	34.08	7th of 10	2.84	6th of 10
BasicVSR++† [6]	39.40	4th of 10	38.24	2nd of 10	35.55	6th of 10	31.72	7th of 10	26.78	9th of 10	34.34	6th of 10	7.41	2nd of 10
VRT† [26]	39.55	2nd of 10	38.12	4th of 10	34.82	8th of 10	30.77	8th of 10	26.01	10th of 10	33.86	8th of 10	0.05	10th of 10
Real-ESRGAN [46]	29.98	10th of 10	28.27	10th of 10	28.04	10th of 10	27.52	10th of 10	27.63	8th of 10	28.29	10th of 10	0.24	8th of 10
UDVD [40]	31.15	9th of 10	30.72	9th of 10	30.23	9th of 10	29.10	9th of 10	27.63	7th of 10	29.77	9th of 10	0.16	9th of 10
MF2F [14]	39.09	6th of 10	38.20	3rd of 10	36.36	1st of 10	33.57	2nd of 10	29.04	4th of 10	35.25	3rd of 10	4.62	5th of 10
Ours - RFCVD	40.35	1st of 10	38.60	1st of 10	36.28	2nd of 10	33.86	1st of 10	31.12	1st of 10	36.04	1st of 10	31.66	1st of 10

Table 3. 在 CRVD（sRGB）基准测试集上的视频去噪结果。我们的方法不仅在整体性能上表现最佳，同时在速度上也比第二快的方法快了四倍。更多关于 SSIM 和 LPIPS 指标的结果请参见补充材料，我们的方法在这些指标上同样排名第一。

	film grain noise w/ AV1 compression						spatially correlated noise w/ H.265 compression						Speed	
	PSNR (higher PSNR is better)	rank	SSIM (higher SSIM is better)	rank	LPIPS (lower LPIPS is better)	rank	PSNR (higher PSNR is better)	rank	SSIM (higher SSIM is better)	rank	LPIPS (lower LPIPS is better)	rank	FPS (higher FPS is better)	rank
SID† [7]	27.05	5th of 7	0.664	5th of 7	0.293	4th of 7	28.44	3rd of 7	0.771	4th of 7	0.282	5th of 7	15.00	3rd of 7
NAFNet† [9]	27.16	4th of 7	0.672	4th of 7	0.294	5th of 7	28.40	4th of 7	0.764	6th of 7	0.257	3rd of 7	3.812	6th of 7
FastDVDNet† [42]	27.39	3rd of 7	0.686	3rd of 7	0.272	3rd of 7	27.92	6th of 7	0.769	5th of 7	0.296	6th of 7	12.09	4th of 7
TOFlow† [49]	28.15	2nd of 7	0.750	2nd of 7	0.220	1st of 7	28.39	5th of 7	0.788	3rd of 7	0.258	4th of 7	5.665	5th of 7
BasicVSR++† [6]	26.90	6th of 7	0.651	6th of 7	0.313	6th of 7	27.48	7th of 7	0.728	7th of 7	0.358	7th of 7	15.32	2nd of 7
VRT† [26]	26.55	7th of 7	0.629	7th of 7	0.331	7th of 7	28.78	2nd of 7	0.803	2nd of 7	0.206	1st of 7	0.105	7th of 7
Ours - RFCVD	28.59	1st of 7	0.774	1st of 7	0.247	2nd of 7	28.93	1st of 7	0.808	1st of 7	0.239	2nd of 7	69.73	1st of 7

Table 4. 在训练时使用加性高斯白噪声（AWGN）结合 H.264 转码，测试时采用两种不同的退化方案，评估方法的泛化能力。由于我们的方法基于经典方法，其去噪过程仅涉及少量参数，因此天然具有较小的领域差异。

4.3. 消融实验

我们最初关注的一个核心问题是：锚帧的选择对性能是否有显著影响。毕竟，I 帧的噪声特性通常与 P 帧和 B 帧有显著差异。尽管如此，如 Tab. 2（上）所示，我们在整个实验部分始终选用第一帧作为锚帧，并验证即便使用中间帧或最后一帧，结论依然保持一致。如 Tab. 2（下）所示，我们还发现，引入 $\mathcal{L}_{cstsy}$ 有助于提供更一致的去噪体验。

我们将经典去噪技术融入机器学习管道的关键设计之一在于：模型需要能够处理空间变化的噪声（spatially-varying noise）。为了验证该设计决策的重要性，我们训练了一个仅预测一组标量去噪参数的消融模型。正如预期，并且如 Tab. 6 所示，该“w/o spat. varying”实验在去噪质量上出现了明显下降。如 Fig. 7 所示，该消融模型生成的图像去噪结果更为模糊，原因在于其难以建模信号相关噪声。

为了更好地支持 $\mathcal{P}(\cdot;\theta)$ 预测去噪参数，我们不仅向其输入（融合后的）图像帧，还提供图像梯度信息以及对齐帧中各像素是否有效的掩码。为测试这一设计的有效性，我们训练了两个消融模型：一个不输入图像梯度（“w/o img. gradients”），另一个不输入对齐掩码（“w/o align. mask”）。如 Tab. 6 所示，这些上下文信息确实对提升性能具有积极作用。

我们认为，对图像噪声特性的分析是低层图像处理与高层语义理解的结合过程。因为要准确分辨噪声与真实纹理之间的差异，首先需要理解图像所表达的内容。因此，我们在超网络中引入了一个分类主干网络 $\mathcal{B}$ 来预测噪声特征 θ，并通过一个去除该主干的消融实验验证了此假设。如 Tab. 6 所示，该“w/o backbone $\mathcal{B}$”实验确实表现出显著的性能下降。如 Fig. 7 所示，缺失该主干网络会导致噪声无法被完全移除。

最后，我们不仅对上述主干网络 $\mathcal{B}$ 进行了微调（fine-tuning），还通过 NPA 方法 [35] 来稳定超网络的训练。为分析这些策略的效果，我们分别训练了两个消融实验：“w/o fine-tuning $\mathcal{B}$”和“w/o NPA”。如 Tab. 6 所示，这两项改动都对最终去噪质量起到了积极的作用。

4.4. 局限性

尽管我们的噪声特征 θ 及其后续通过 $\mathcal{P}(\cdot;\theta)$ 预测的去噪参数在实验部分已表现出合理的效果，但仍远非完美。具体而言，如 Tab. 5 所示，我们对预测的去噪参数进行了基于真实标注的微调，发现这一假设的“oracle”能够显著提升我们的结果。尽管如此，弥合这一差距仍然留待未来研究解决。

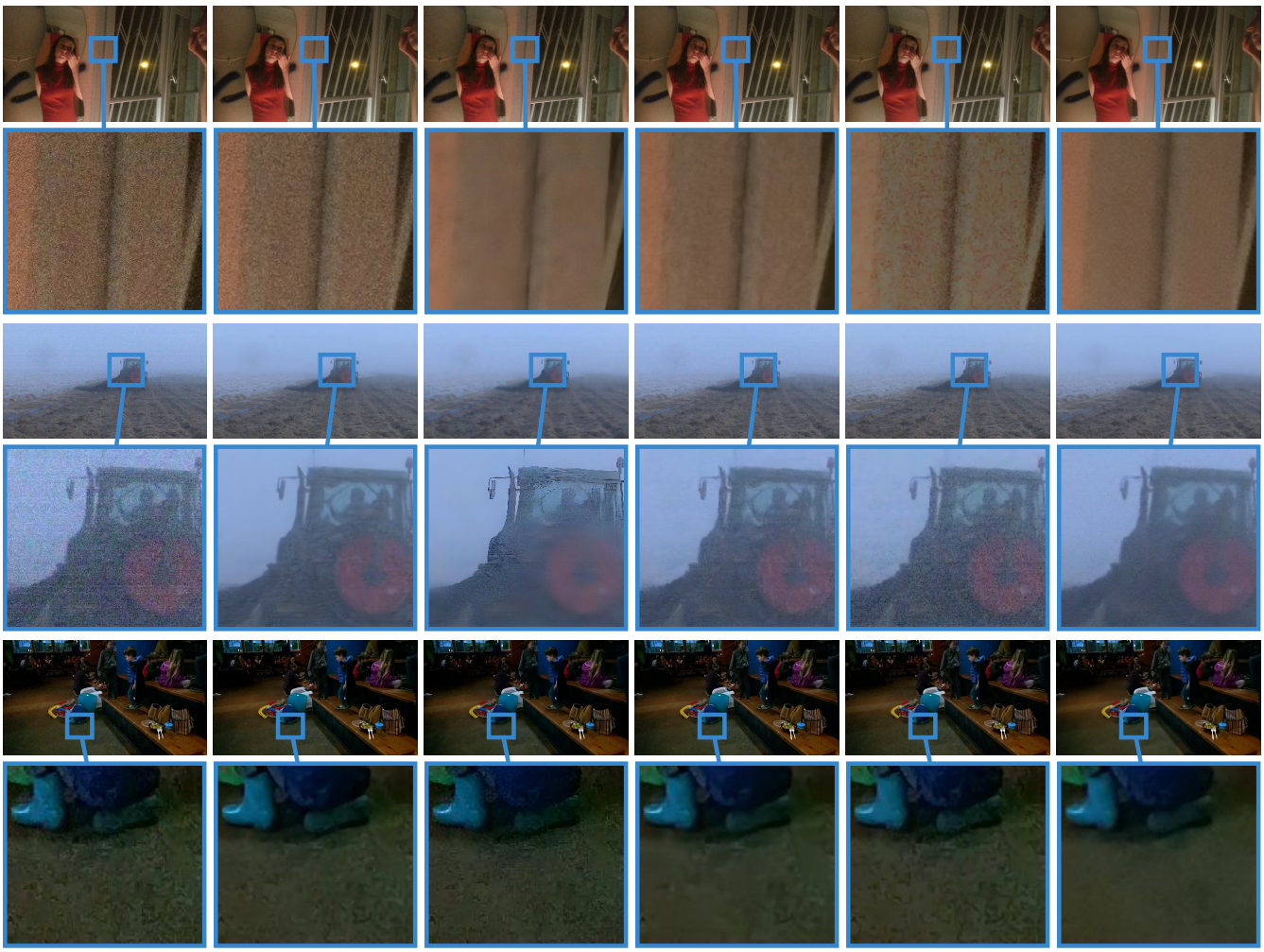


Figure 6. 视频去噪的示例静帧结果。由于篇幅限制，我们仅展示了部分具有代表性的方法的结果。完整的视频对比结果请参阅附录。我们由衷感谢 Amit Zinman（第一行）、Robert Kjettrup（中间一行）以及一位匿名艺术家（最后一行）愿意提供测试素材。

	PSNR delta (higher PSNR is better)	SSIM delta (higher SSIM is better)	LPIPS delta (lower LPIPS is better)
Ours	36.04 –	0.9472 –	0.0763 –
fine-tuned on GT	36.75 + 0.70	0.9494+ 0.0022	0.0591- 0.0172

Table 5. 在 CRVD 基准上的真实数据上微调我们估计的去噪参数表明：我们的估计结果已经较为准确，但如果能够访问一个理想的“oracle”，结果仍有进一步提升的空间。

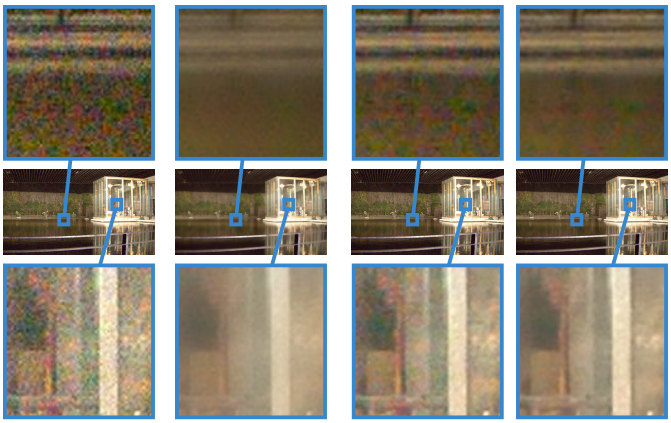
在噪声特征完全已知的情况下，例如针对特定摄像机型号和受限编码器设计去噪器时，任何合理的深度学习方法都可能优于我们的方法。并且在噪声特征固定的条件下，相关方法的速度甚至可能更快。然而，此类方法在鲁棒性和控制能力方面会受到限制。

最后，更为实际的一点是，目前主流推理生态系统对超网络的支持尚不完善。具体而言，像 CoreML、WinML 和 OpenVINO 这类流行库通常要求模型权重是固定的，而超网络显然不满足这一条件。因此，虽然我们的方法在 PyTorch 等框架中易于实现，但在生产环境中部署仍不够简便。

5. 结论

我们展示了传统去噪技术在现代机器学习领域依然具有重要价值。我们发现这些方法不仅稳健且高效，同时具备可控性，这一点在许多视频去噪应用中尤为关键，因为创意专业人士希望在去噪过程中表达其艺术诉求。为了弥合经典去噪方法与现代技术之间的差距，我们训练了一个模型，用以估计传统去噪方法针对特定输入视频所需的各类参数，而这些参数的手动调节通常既繁琐又需要一定的专业技能。

此外，为了充分利用传统去噪技术的计算效率，



Input w/o spat. varying w/o backbone $\mathcal{B}$ Ours

Figure 7. 对 CRVD 数据集代表性样本的两组关键消融实验结果进行可视化对比。

	PSNR delta (higher PSNR is better)	SSIM delta (higher SSIM is better)	LPIPS delta (lower LPIPS is better)
Ours	36.04 –	0.9472 –	0.0763 –
w/o spat. varying	33.62 - 2.42	0.9359 - 0.0112	0.1184 + 0.0421
w/o img. gradients	35.61 - 0.44	0.9469 - 0.0003	0.0756 - 0.0007
w/o align. mask	35.26 - 0.78	0.9468 - 0.0003	0.0740 - 0.0023
w/o backbone $\mathcal{B}$	34.38 - 1.67	0.9059 - 0.0412	0.1444 + 0.0680
w/o fine-tuning $\mathcal{B}$	35.71 - 0.34	0.9464 - 0.0008	0.0732 - 0.0032
w/o NPA [35]	35.83 - 0.21	0.9453 - 0.0018	0.0814 + 0.0051

Table 6. 开展了多项消融实验，有关各消融实验的具体背景与详细信息，请参见第 4.3 节。

我们的方法将噪声分析与去噪过程分离，避免了机器学习视频去噪方法中常见的重复估计噪声分布的问题。最后，我们发现了一种基于加性高斯白噪声（AWGN）并经过 H.264 视频转码的简洁而高效的退化流程。该流程确保了噪声具有时间相关性，这符合现实世界中的常见情况。我们不仅验证了该流程对本方法的有效性，还成功地利用它对现有模型进行了再训练并取得了性能提升。

致谢

This work was supported in part by the National Natural Science Foundation of China (62306153, 62225604), the Fundamental Research Funds for the Central Universities (Nankai University, 070-63243143), the Natural Science Foundation of Tianjin, China (24JCJQJC00020), and Shenzhen Science and Technology Program (JCYJ20240813114237048). The computational devices is partly supported by the Supercomputing Center of Nankai University (NKSC).

本工作部分得到了国家自然科学基金（项目编号：62306153，62225604）、中央高校基础科研业务费（南开大学，070-63243143）、天津市自然科学基金（24JCJQJC00020）以及深圳市科技计划（JCYJ20240813114237048）的资助。计算设备部分支持来自南开大学超级计算中心（NKSC）。

References

[1] Eric P Bennett and Leonard McMillan. Video enhancement using per-pixel virtual exposures. In ACM SIGGRAPH 2005 Papers, 2005. 2

[2] Goutam Bhat, Michaël Gharbi, Jiawen Chen, Luc Van Gool, and Zhihao Xia. Self-supervised burst super-resolution. In ICCV, pages 10571–10580. IEEE, 2023. 4

[3] Peter J Burt and Edward H Adelson. The laplacian pyramid as a compact image code. In Readings in computer vision, 1987. 2

[4] Jiezhang Cao, Yawei Li, Kai Zhang, and Luc Van Gool. Video super-resolution transformer. arXiv:2106.06847, 2021. 2

[5] Kelvin CK Chan, Xintao Wang, Ke Yu, Chao Dong, and Chen Change Loy. Basicvsr: The search for essential components in video super-resolution and beyond. In CVPR, 2021. 2

[6] Kelvin CK Chan, Shangchen Zhou, Xiangyu Xu, and Chen Change Loy. Basicvsr++: Improving video super-resolution with enhanced propagation and alignment. In CVPR, 2022. 2, 5, 6

[7] Chen Chen, Qifeng Chen, Jia Xu, and Vladlen Koltun. Learning to see in the dark. In CVPR, 2018. 2, 5, 6

[8] Jiawen Chen, Sylvain Paris, and Frédo Durand. Real-time edge-aware image processing with the bilateral grid. ACM TOG, 2007. 2

[9] Liangyu Chen, Xiaojie Chu, Xiangyu Zhang, and Jian Sun. Simple baselines for image restoration. In ECCV, 2022. 2, 5, 6

[10] Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian. Image denoising with block-matching and 3d filtering. In Image Processing, 2006. 2

[11] Kostadin Dabov, Alessandro Foi, and Karen Egiazarian. Video denoising by sparse 3d transform domain collaborative filtering. EURASIP, 2007.

[12] Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian. Image denoising by sparse 3-d transform-domain collaborative filtering. IEEE TIP, 2007. 2

[13] Axel Davy, Thibaud Ehret, Jean-Michel Morel, Pablo Arias, and Gabriele Facciolo. A non-local cnn for video denoising. In ICIP, 2019. 2

[14] Valéry Dewil, Jérémy Anger, Axel Davy, Thibaud Ehret, Gabriele Facciolo, and Pablo Arias. Self-supervised training for blind multi-frame video denoising. In WACV, 2021. 1, 2, 4, 5, 6, 7

[15] Charles Dugas, Yoshua Bengio, François Bélisle, Claude Nadeau, and René Garcia. Incorporating second-order functional knowledge for better option pricing. In NIPS, 2000. 4

[16] Jana Ehmann, Lun-Cheng Chu, Sung-Fang Tsai, and Chia-Kai Liang. Real-time video denoising on mobile phones. In ICIP, 2018. 2

[17] Thibaud Ehret, Axel Davy, Jean-Michel Morel, Gabriele Facciolo, and Pablo Arias. Model-blind video denoising via frame-to-frame training. In CVPR, 2019. 2

[18] Ruturaj G Gavaskar and Kunal N Chaudhury. Fast adaptive bilateral filtering. IEEE TIP, 2018. 2

[19] David Ha, Andrew M. Dai, and Quoc V. Le. Hypernetworks. arXiv:1609.09106, 2016. 3

[20] Samuel W Hasinoff, Dillon Sharlet, Ryan Geiss, Andrew Adams, Jonathan T Barron, Florian Kainz, Jiawen Chen, and Marc Levoy. Burst photography for high dynamic range and low-light imaging on mobile cameras. ACM TOG, 2016. 2, 4

[21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In ICCV, 2015. 4

[22] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In ICLR, 2015. 5

[23] Samuli Laine, Tero Karras, Jaakko Lehtinen, and Timo Aila. High-quality self-supervised deep image denoising. In NeurIPS, 2019. 2

[24] Seunghwan Lee, Donghyeon Cho, Jiwon Kim, and Tae Hyun Kim. Restore from restored: Video restoration with pseudo clean video. In CVPR, 2021. 2

[25] Jaakko Lehtinen, Jacob Munkberg, Jon Hasselgren, Samuli Laine, Tero Karras, Miika Aittala, and Timo Aila. Noise2noise: Learning image restoration without clean data. In ICML, 2018. 2

[26] Jingyun Liang, Jiezhang Cao, Yuchen Fan, Kai Zhang, Rakesh Ranjan, Yawei Li, Radu Timofte, and Luc Van Gool. Vrt: A video restoration transformer. IEEE TIP, 2024. 1, 2, 5, 6, 7

[27] Orly Liba, Kiran Murthy, Yun-Ta Tsai, Tim Brooks, Tianfan Xue, Nikhil Karnad, Qiurui He, Jonathan T Barron, Dillon Sharlet, Ryan Geiss, et al. Handheld mobile photography in very low light. ACM TOG, 2019. 2

[28] Ce Liu and William T Freeman. A high-quality video denoising algorithm based on reliable motion estimation. In ECCV, 2010. 2

[29] Ding Liu, Zhaowen Wang, Yuchen Fan, Xianming Liu, Zhangyang Wang, Shiyu Chang, and Thomas Huang. Robust video super-resolution with learned temporal dynamics. In ICCV, 2017. 2

[30] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In CVPR, 2022. 3, 4

[31] Matteo Maggioni, Vladimir Katkovnik, Karen Egiazarian, and Alessandro Foi. Nonlocal transform-domain filter for volumetric data denoising and reconstruction. IEEE TIP, 2012. 2

[32] Matteo Maggioni, Yibin Huang, Cheng Li, Shuai Xiao, Zhongqian Fu, and Fenglong Song. Efficient multi-stage video denoising with recurrent spatio-temporal fusion. In CVPR, 2021. 2

[33] Seungjun Nah, Sungyong Baik, Seokil Hong, Gyeongsik Moon, Sanghyun Son, Radu Timofte, and Kyoung Mu Lee. NTIRE 2019 challenge on video deblurring and super-resolution: Dataset and study. In CVPRW, 2019. 4, 5

[34] Alasdair Newson, Noura Faraj, Bruno Galerne, and Julie Delon. Realistic film grain rendering. IPOL, 2017. 5

[35] Jose Javier Gonzalez Ortiz, John V. Guttag, and Adrian V. Dalca. Non-proportional parametrizations for stable hypernetwork learning. arXiv:2304.07645, 2023. 3, 6, 8

[36] Sylvain Paris, Pierre Kornprobst, Jack Tumblin, Frédo Durand, et al. Bilateral filtering: Theory and applications. Foundations and Trends® in Computer Graphics and Vision, 2009. 2

[37] Andrea Petreto, Thomas Romera, Florian Lemaitre, Ian Masliah, Boris Gaillard, Manuel Bouyer, Quentin L Meunier, and Lionel Lacassagne. A new real-time embedded video denoising algorithm. In DASIP, 2019. 2

[38] Anurag Ranjan and Michael J. Black. Optical flow estimation using a spatial pyramid network. In CVPR, 2017. 3

[39] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In MICCAI, 2015. 5

[40] Dev Yashpal Sheth, Sreyas Mohan, Joshua L Vincent, Ramon Manzorro, Peter A Crozier, Mitesh M Khapra, Eero P Simoncelli, and Carlos Fernandez-Granda. Unsupervised deep video denoising. In ICCV, 2021. 1, 2, 4, 5, 6, 7

[41] Maitreya Suin and AN Rajagopalan. Gated spatio-temporal attention-guided video deblurring. In CVPR, 2021. 2

[42] Matias Tassano, Julie Delon, and Thomas Veit. Fastdvdnet: Towards real-time deep video denoising without flow estimation. In CVPR, 2020. 2, 5, 6

[43] Carlo Tomasi and Roberto Manduchi. Bilateral filtering for gray and color images. In ICCV, 1998. 2

[44] Gregory Vaksman, Michael Elad, and Peyman Milanfar. Patch craft: Video denoising by deep modeling and patch matching. In ICCV, 2021. 2

[45] Xintao Wang, Kelvin CK Chan, Ke Yu, Chao Dong, and Chen Change Loy. Edvr: Video restoration with enhanced deformable convolutional networks. In CVPRW, 2019. 2

[46] Xintao Wang, Liangbin Xie, Chao Dong, and Ying Shan. Real-esrgan: Training real-world blind super-resolution with pure synthetic data. In ICCV Workshop, 2021. 1, 2, 4, 5, 6, 7

[47] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. IEEE TIP, 2004. 5

[48] Bartlomiej Wronski, Ignacio Garcia-Dorado, Manfred Ernst, Damien Kelly, Michael Krainin, Chia-Kai

Liang, Marc Levoy, and Peyman Milanfar. Handheld multi-frame super-resolution. ACM TOG, 2019. 2

[49] Tianfan Xue, Baian Chen, Jiajun Wu, Donglai Wei, and William T Freeman. Video enhancement with task-oriented flow. IJCV, 2019. 2, 5, 6

[50] Huanjing Yue, Cong Cao, Lei Liao, Ronghe Chu, and Jingyu Yang. Supervised raw video denoising with a benchmark dataset on dynamic scenes. In CVPR, 2020. 2, 4, 5

[51] Kai Zhang, Jingyun Liang, Luc Van Gool, and Radu Timofte. Designing a practical degradation model for deep blind image super-resolution. In CVPR, 2021. 2

[52] Kai Zhang, Yawei Li, Jingyun Liang, Jiezhang Cao, Yulun Zhang, Hao Tang, Deng-Ping Fan, Radu Timofte, and Luc Van Gool. Practical blind image denoising via swin-conv-unet and data synthesis. MIR, 2023. 2

[53] Ming Zhang and Bahadir K. Gunturk. Multiresolution bilateral filtering for image denoising. IEEE TIP, 2008. 4

[54] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In CVPR, 2018. 5