

# 第17讲：时空滤波

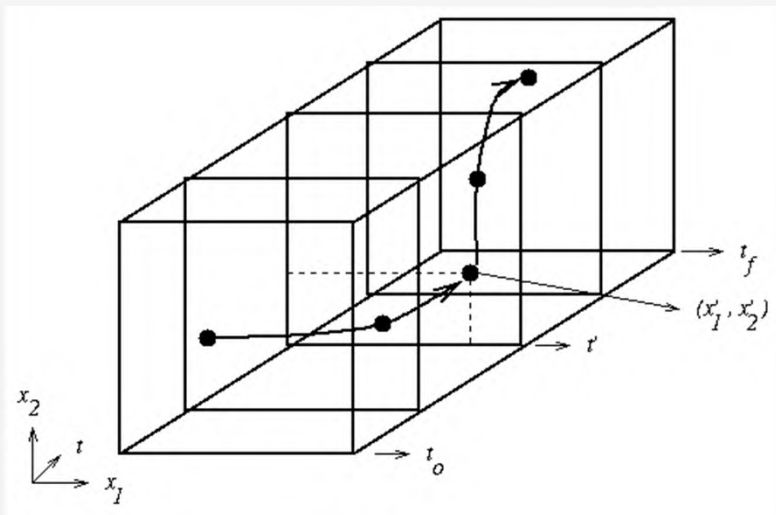
---

- 时空滤波理论
  - 具有匀速运动的视频频谱
  - 运动补偿滤波
  - 无抗混叠滤波的下采样
- 视频格式转换
  - 图像长宽比转换
  - 去隔行
  - 帧/场速率转换
- 时空噪声滤波
  - 运动自适应滤波
  - 运动补偿滤波
- 多帧复原

# 时空滤波理论

- 给定**运动轨迹** $c(t; x_1, x_2, t_0)$ ，在时刻 $t'$ 沿轨迹运动的点 $(x'_1, x'_2)$ 的速度可以由下式定义：

$$\mathbf{v}(x'_1, x'_2, t') = \frac{d\mathbf{c}}{dt}(t; x_1, x_2, t_0)|_{t=t'}$$



在时间 $t'$ 处通过像素  $(x'_1, x'_2)$  的运动轨迹。

- 运动估计/补偿的基本假设是像素的**强度沿着运动轨迹保持不变**，即施加了对**视频局部时空频谱的约束**。

# 视频的时空频谱

- 具有**恒定速度**( $v_1, v_2$ )的**全局平移**可以被建模为:

$$s_c(x_1, x_2, t) = s_c(x_1 - v_1 t, x_2 - v_2 t, 0) \equiv s_0(x_1 - v_1 t, x_2 - v_2 t)$$

其中 $s_0(x_1, x_2)$ 表示在 $t_0 = 0$ 时参考的2D强度函数。

- FT的任意时空信号的**支撑区域**占据了整个 $F_1, F_2, F_t$ 空间:

$$S_c(F_1, F_2, F_t) = \iiint s_c(x_1, x_2, t) e^{-j2\pi(F_1 x_1 + F_2 x_2 + F_t t)} dx_1 dx_2 dt$$

- 对于**具有恒速平移运动**的视频, 有:

$$S_c(F_1, F_2, F_t) = \iiint s_0(x_1 - v_1 t, x_2 - v_2 t) e^{-j2\pi(F_1 x_1 + F_2 x_2 + F_t t)} dx_1 dx_2 dt$$

- 变量置换:  $x'_i = x_i - v_i t$ ,  $i = 1, 2$ , 可得:

$$S_c(F_1, F_2, F_t) = \iint s_0(x'_1, x'_2) e^{-j2\pi(F_1 x'_1 + F_2 x'_2)} dx'_1 dx'_2 \int e^{-j2\pi(F_1 v_1 + F_2 v_2 + F_t)t} dt$$

可以简化为:

$$S_c(F_1, F_2, F_t) = S_0(F_1, F_2) \delta(F_1 v_1 + F_2 v_2 + F_t)$$

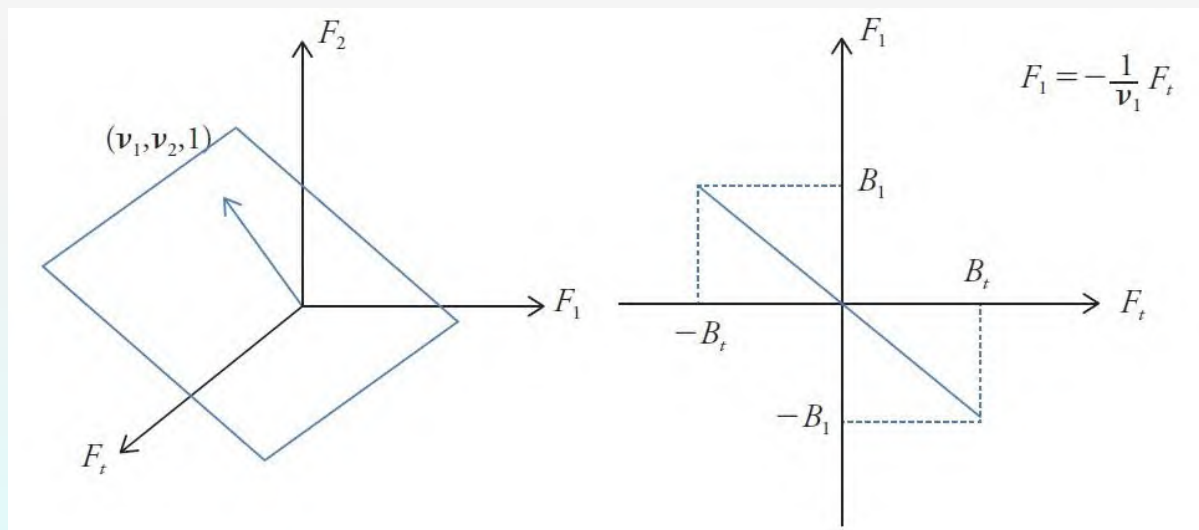
其中  $S_0(F_1, F_2)$  是 $s_0(x_1, x_2)$ 的二维傅立叶变换,  $\delta(\cdot)$ 为狄拉克函数。 3

## 视频的时空频谱 (cont'd)

- 只有当它的参数  $\mathbf{F}^T \mathbf{v} = 0$  时,  $\delta$  函数才是非零的。因此,  $\delta$  函数将  $S_c(F_1, F_2, F_t)$  的支撑区域限制在  $\mathbb{R}^3$  给出的平面上:

$$F_1 v_1 + F_2 v_2 + F_t = 0$$

其通过原点, 并且与矢量  $\mathbf{v}$  正交。



对于:  $|F_t| > B_t$ ,  $S_c(F_1, F_2, F_t) = 0$ , 其中:

$$B_t = B_1 v_1 + B_2 v_2$$

简言之,  $S_c(F_1, F_2, F_t)$  的支撑区域到  $(F_1, F_t)$  平面投影定义为:

$$F_1 v_1 + F_t = 0,$$

- 这个结果是很直观的。由于参考帧足以根据运动模型的性质确定所有将来的帧, 所以参考帧的傅立叶变换将足以表示视频的整个时空频谱。



# 运动补偿滤波

- 运动补偿时空滤波器的脉冲响应由下式给出：

$$h(x_1, x_2, t) = h_1(t) \delta(x_1 - v_1 t, x_2 - v_2 t)$$

其中 $h_1(t)$ 是沿着运动轨迹所使用的1D滤波器的脉冲响应， $v_1, v_2$ 表示速度（运动矢量）估计。

- 因此，沿着恒速运动轨迹的线性移不变滤波是：

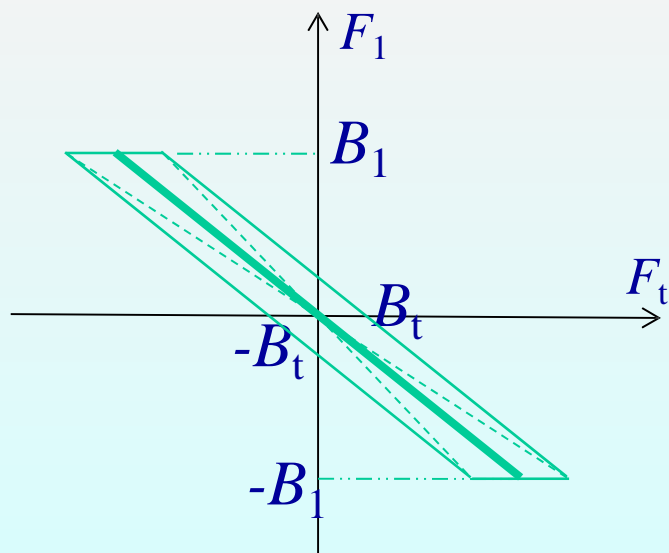
$$\begin{aligned} y(x_1, x_2, t) &= \iiint h_1(\tau) \delta(z_1 - v_1 \tau, z_2 - v_2 \tau) s_c(x_1 - z_1, x_2 - z_2, t - \tau) dz_1 dz_2 d\tau \\ &= \int h_1(\tau) s_c(x_1 - v_1 \tau, x_2 - v_2 \tau, t - \tau) d\tau \end{aligned}$$

- 运动补偿滤波器的频率响应由三维傅立叶变换给出：

$$\begin{aligned} H(F_1, F_2, F_t) &= \iiint h_1(t) \delta(x_1 - v_1 t, x_2 - v_2 t) e^{-j2\pi(F_1 x_1 + F_2 x_2 + F_t t)} dx_1 dx_2 dt \\ &= \int h_1(t) e^{-j2\pi(F_1 v_1 + F_2 v_2 + F_t) t} dt = H_1(F_1 v_1 + F_2 v_2 + F_t) \end{aligned}$$

## 运动补偿滤波 (cont'd)

- 频率响应  $H(F_1, F_2, F_t)$  在平面  $F_1 v_1 + F_2 v_2 + F_t = F$  上是常数。
- 如果输入的频谱被限制在平面  $F_1 v_1 + F_2 v_2 + F_t = 0$ ，则滤波效果就是简单地乘以  $H_1(0)$ 。
- 投影到  $(F_1, F_t)$  平面上的3D时空低通滤波器的通带为平行四边形：  
$$H_p(F_1, F_t) = \begin{cases} 1 & -B_1 < F_1 < B_1 \quad -v_1 F_1 - B_t < F_t < -v_1 F_1 + B_t \\ 0 & \text{其它} \end{cases}$$



通带宽度为  $2B_t$ ，斜率为  $-v_1$ 。实线是具有匹配速度的输入频谱。虚线表示容差范围  $\Delta$ 。

当  $v_1$  与输入视频的速度匹配时，可实现正确的运动补偿。

# 下转换

- 定义一个 $M \times M$ 子抽样矩阵 $\mathbf{S} = \mathbf{V}_1^{-1} \mathbf{V}_2$ ，使得 $\mathbf{V}_2 = \mathbf{V}_1 \mathbf{S}$ 。由于 $\mathbf{V}_1$ 总是可逆的，所以总可以定义一个子采样矩阵。然后，输出格 $\Lambda_2$ 的位置可以表示为：

$$\mathbf{y} = \mathbf{V}_1 \mathbf{S} \mathbf{n}, \quad \mathbf{n} \in \mathbf{Z}^M$$

- 例如，根据矩阵的时空子采样

$$\mathbf{S} = \begin{bmatrix} 2 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

对应于2:1隔行，即分别在交替帧中丢弃偶数和奇数列。

- 使用抗混叠滤波进行下转换
- 未使用抗混叠滤波的下转换

# 抗混叠滤波情况下的下转换

- 从ITU-R 601 4:2:2隔行演播室格式到ITU-R 601 4:2:0隔行广播格式的转换需要在**垂直方向以采样因子2进行抽取**。
- 因为输入视频是隔行扫描，所以每场的**简单垂直抽取**都不会保留两场中行与行之间的空间偏移。
- 可以通过使用**奇数长度的滤波器**在每隔一行除去一行之前对其中的一个色度场（垂直方向）进行滤波来解决该问题，并且通过使用**均匀长度滤波器**来处理交替色度场以将行的位置偏移 $1/2$ 。
- 奇数和偶数长度的**滤波器脉冲响应**分别为 $[-29, 0, 88, 138, 88, 0, -29] / 256$ 和 $[1, 7, 7, 1] / 16$ 。
- 此转换中的**亮度通道不会被修改**。

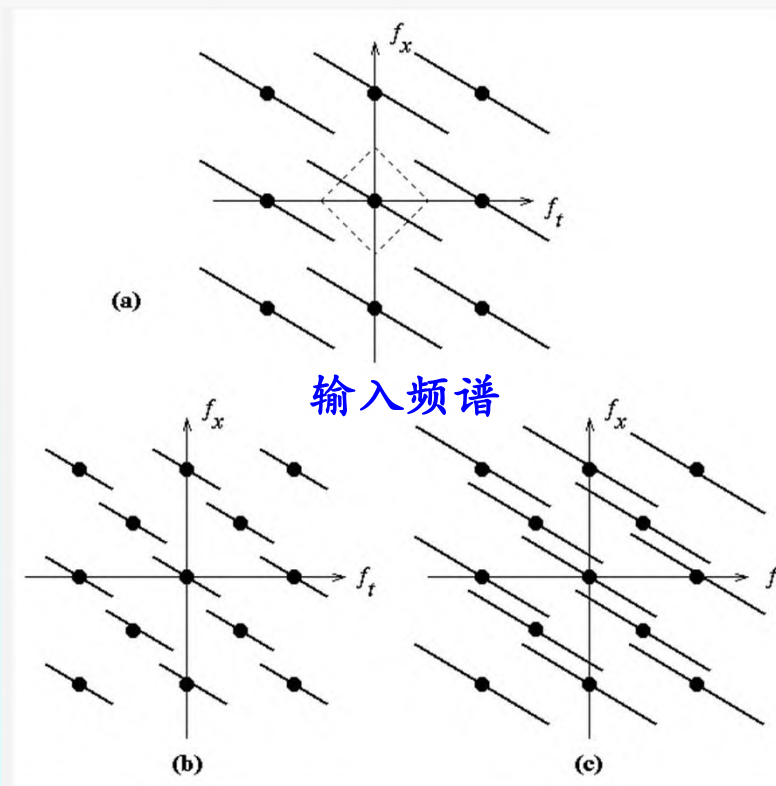
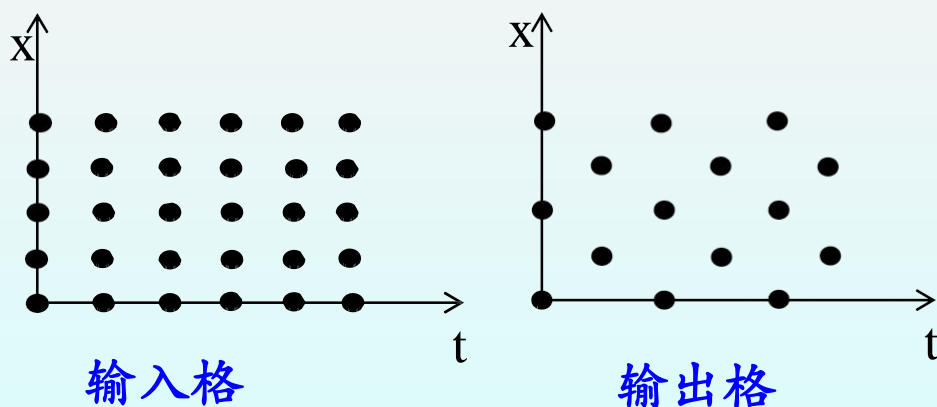


# 无抗混叠滤波情况下的下转换

- 根据子采样矩阵通过对逐行输入格 $V_1$ 进行子采样，获得输出格 $V_2$ 。

$$S = \begin{bmatrix} 2 & 1 \\ 0 & 1 \end{bmatrix}$$

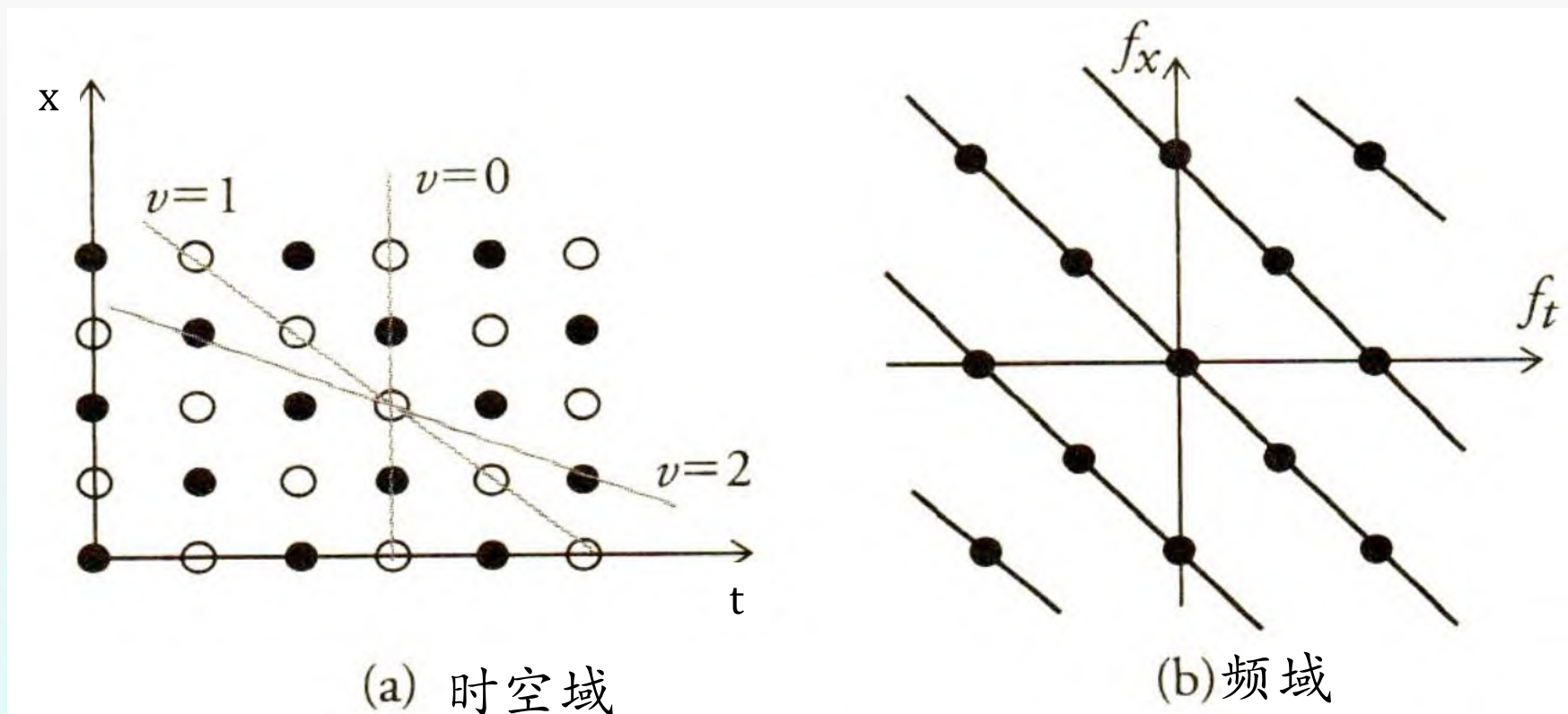
这对应于在交替时间采样中**丢弃偶数和奇数行**。



具有和不具有抗混叠滤波器的  
下采样信号的频谱。

# 无抗混叠滤波情况下的下转换

- 临界速度

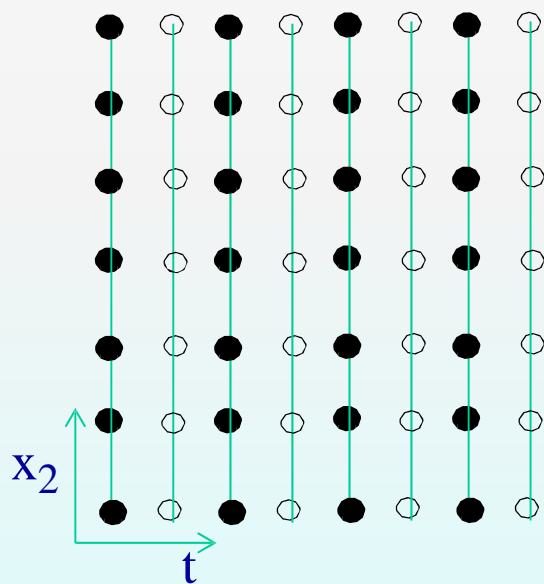


$$f_t = -vf_x$$

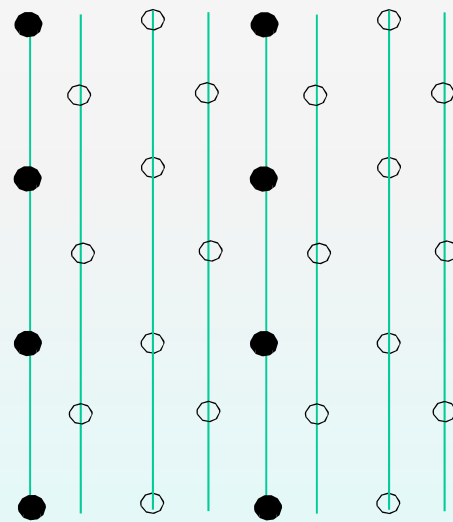
临界速度为  $v_c = 2i + 1$ ,  $i \in \mathbb{Z}$

# 视频格式转换问题

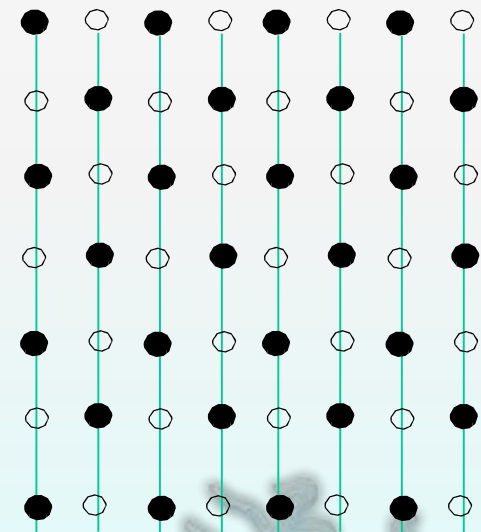
- 帧/场速率转换
- 去隔行



帧率转换



场率转换

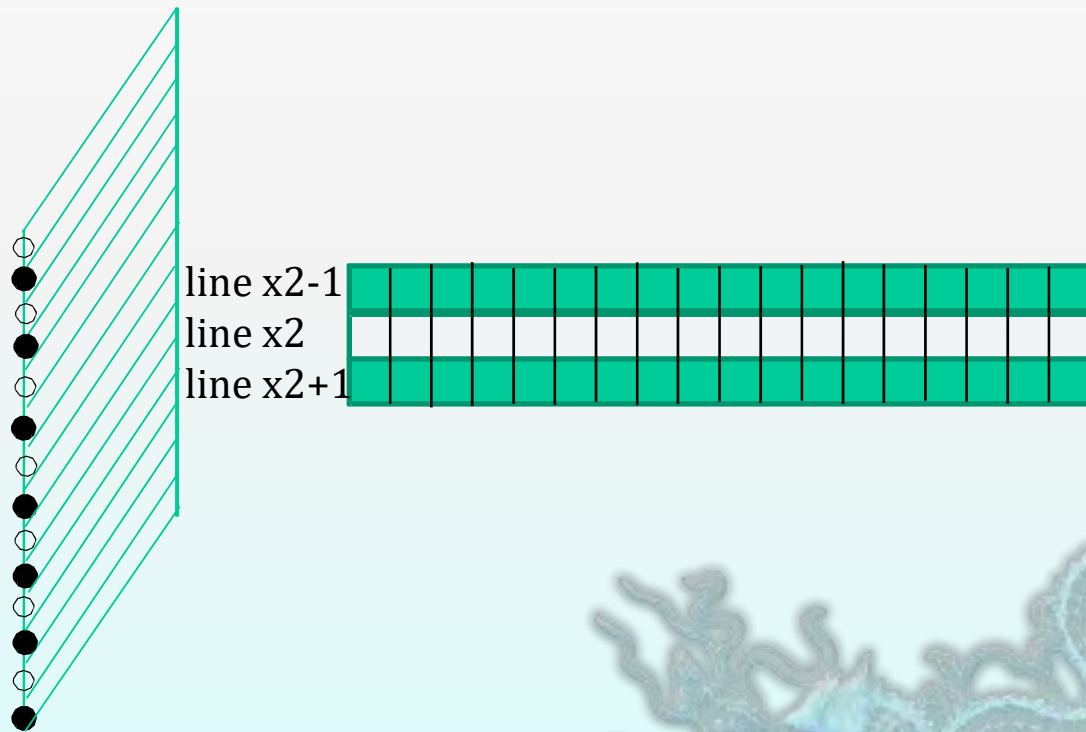


去隔行

# 去隔行

## 从50i/60i 转换到 50p/60p

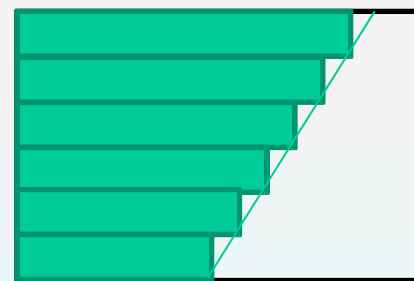
- 场内空域去隔行 (**bob**滤波)
  - 线性插值
  - 边缘定向插值
- 场间时域去隔行 (**weave**滤波)
- 运动 **自适应**
- 运动**补偿**





# 场交织

- **合并**（叠加）偶数和奇数场。
- 在**静止图像区域**最优
- 运动区域中的**抖动损伤**（在快速运动的情况下将更显著）



无抖动



抖动

# 线性插值

场交织



线性插值



# 边缘定向插值

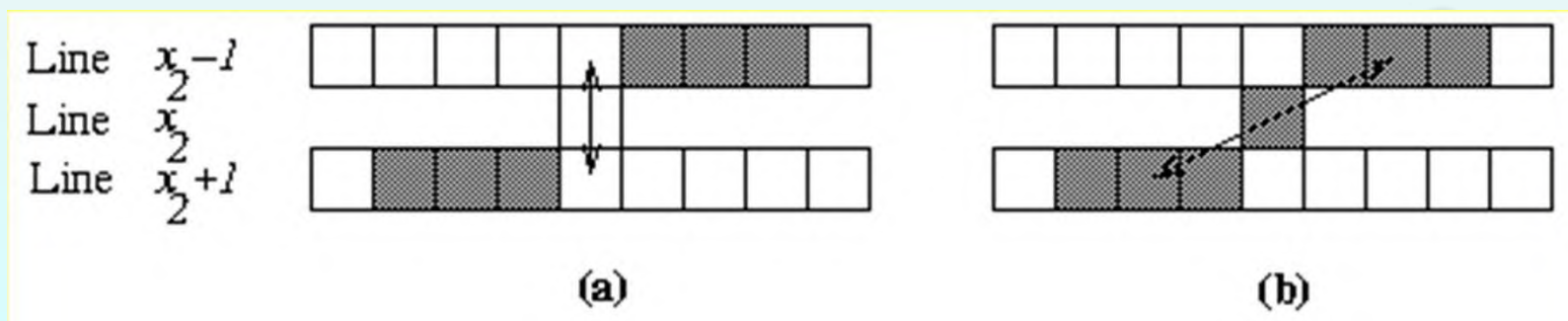
- 场/帧 $t_0$ 中的每行视频被局部地建模为相同场/帧中的**前一行的水平位移版本**:

$$s(x_1 - d/2 + j, x_2 - 1, t_0) = s(x_1 + d/2 + j, x_2 + 1, t_0)$$

其中 $d$  是两个连续偶/奇行之间的局部水平位移。

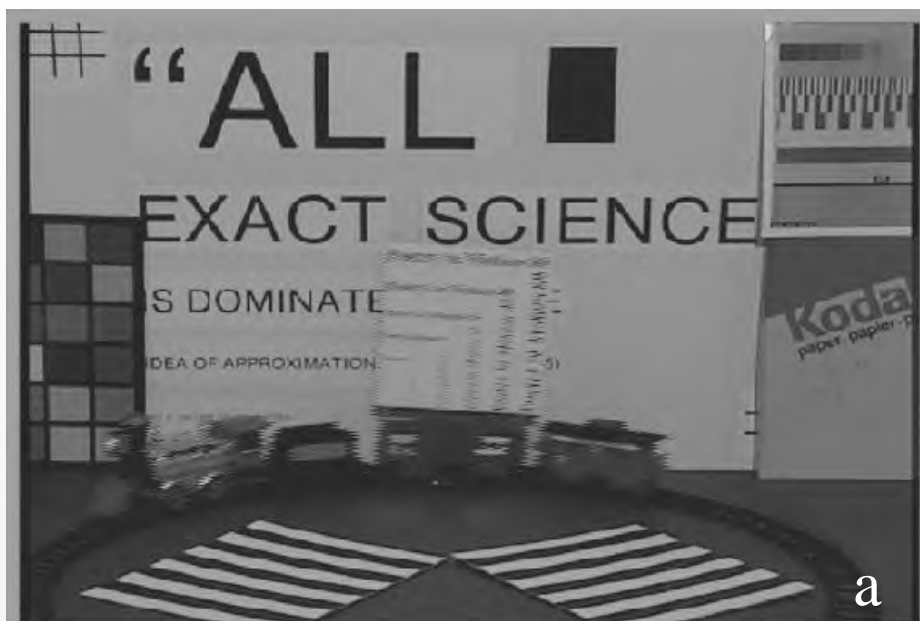
- 通过**对称线段匹配**( $x_1, x_2$ ) 来估计每个像素处的位移 $d$ , 以便最小化绝对误差和 (SAD) :

$$\text{SAD}(d) = \sum_{j=-1}^1 |s(x_1 - d/2 + j, x_2 - 1, t_0) - s(x_1 + d/2 + j, x_2 + 1, t_0)|$$





# 例子



- a) 复合帧
- b) 线性内插
- c) 边缘自适应内插



# 运动自适应去隔行

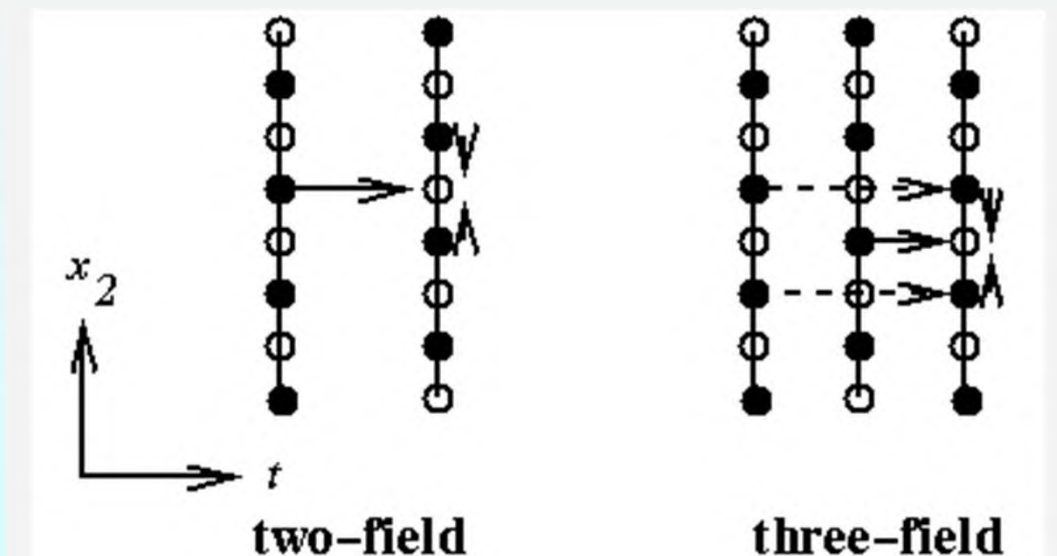
- 3场 Bob - Weave 滤波器:

$$S(x_1, x_2, t_i) = \alpha s\left(x_1 - \frac{d}{2}, x_2 - 1, t_i\right) + \beta s\left(x_1 + \frac{d}{2}, x_2 + 1, t_i\right) + \gamma s(x_1, x_2, t_i - 1)$$

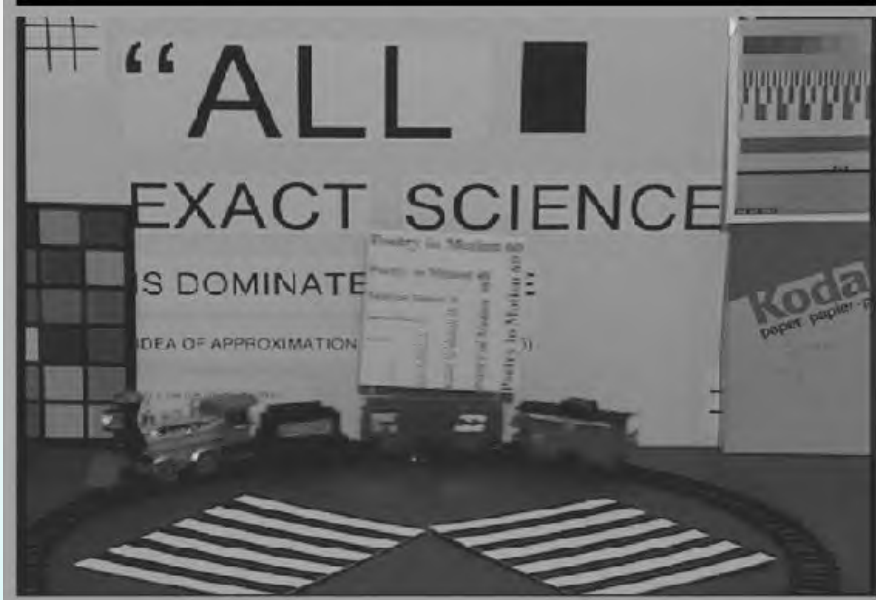
$\alpha = 0.5$      $\beta = 0.5$ ,     $\gamma = 0$     如果检测到运动 (Bob)

$\alpha = 0$ ,     $\beta = 0$      $\gamma = 1$     如果没有检测到运动 (Weave)

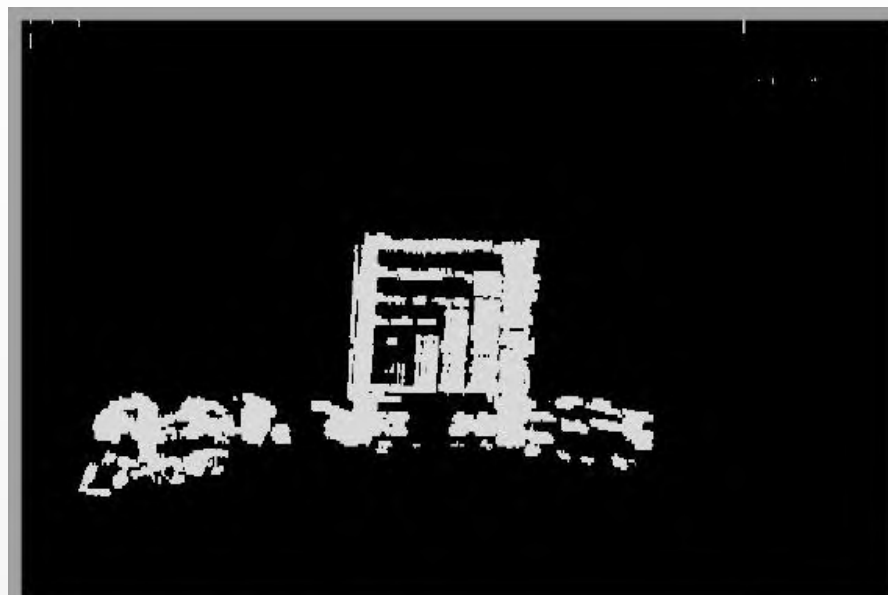
参数d启用边缘自适应帧内插值。



- 运动检测
- 自适应滤波



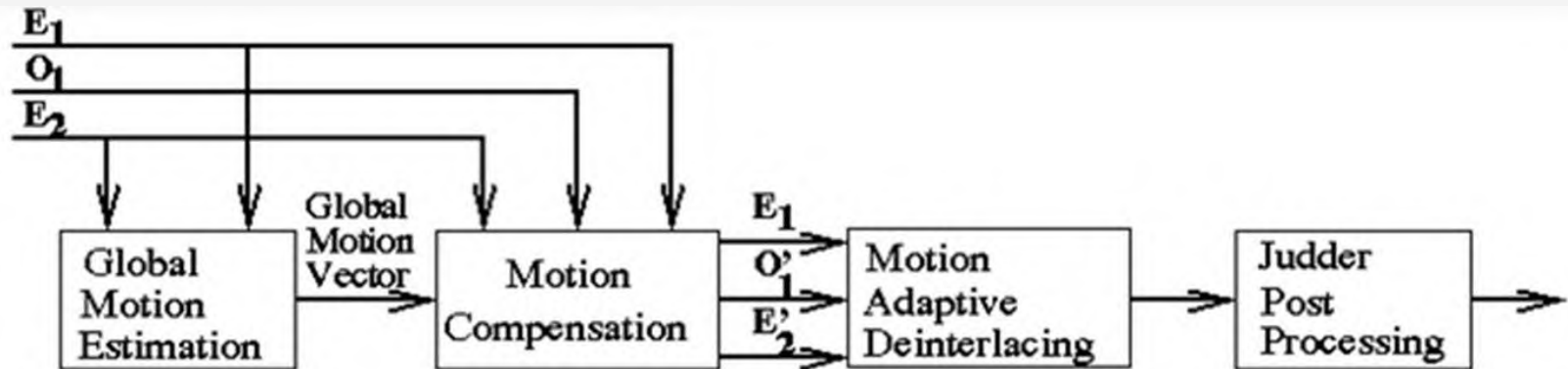
3场运动自适应



4场运动自适应

# 全局运动补偿

- 步骤 1: 估计并补偿全局摄像机运动
- 步骤 2: 使用 Bob-Weave 滤波器



# 示例：手持式摄像机



复合帧

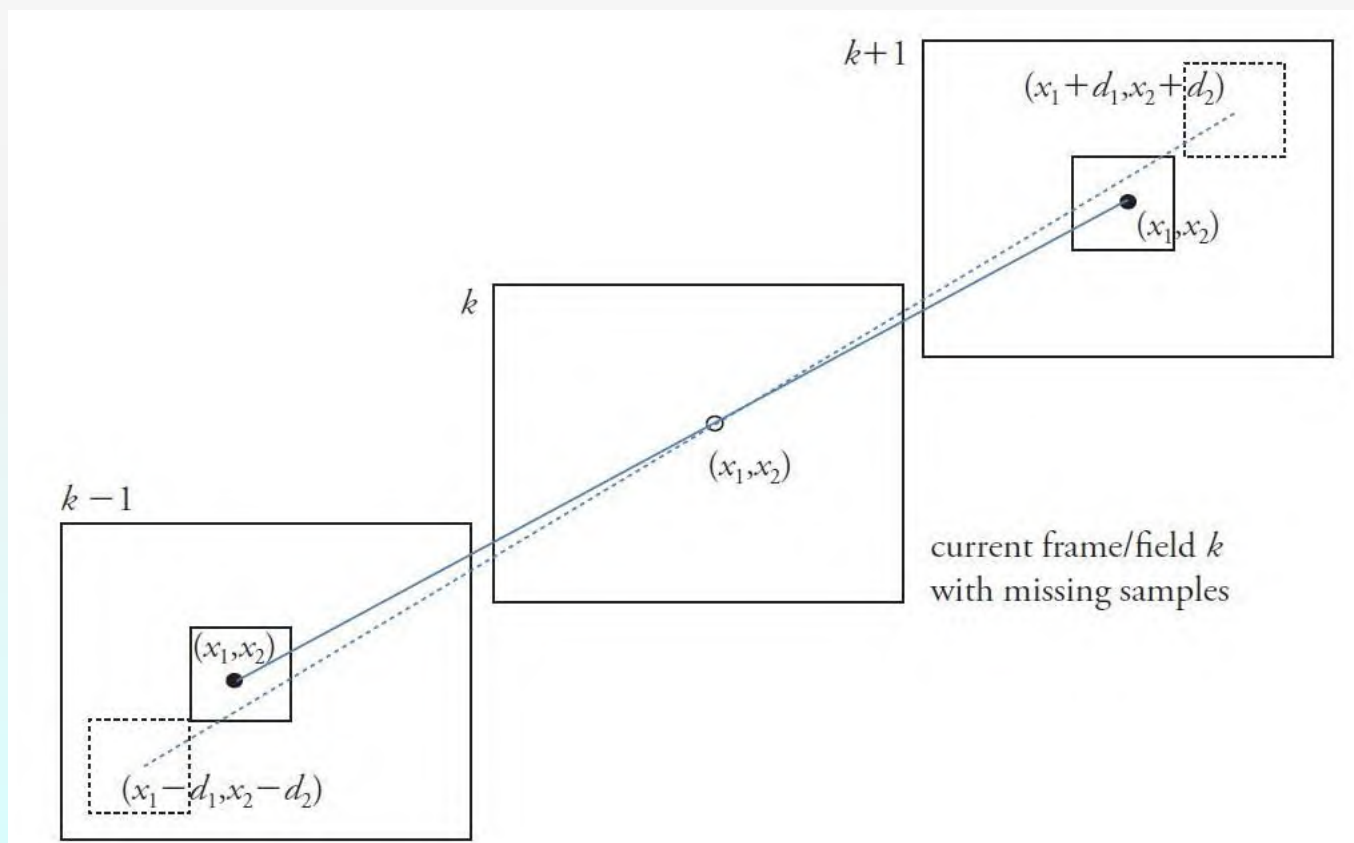


全局运动补偿, 运动自适应



# 运动补偿去隔行

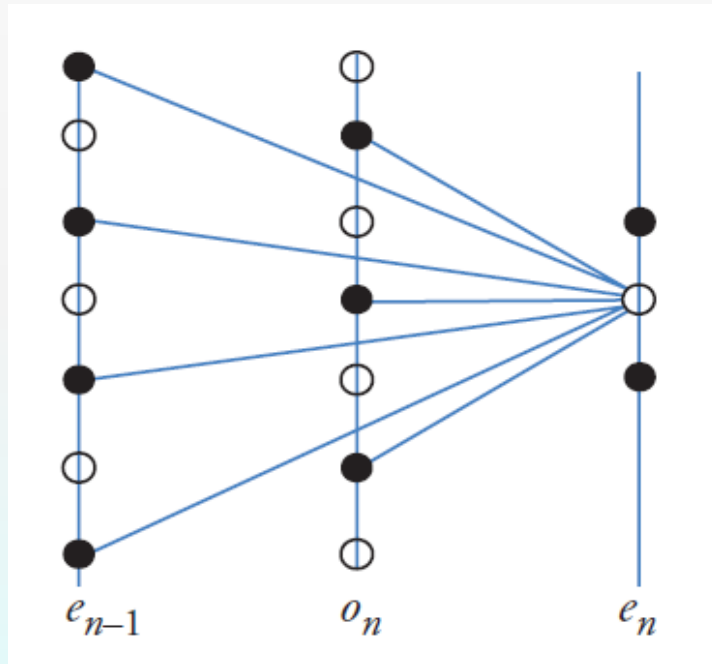
- 缺失样本处的运动估计
  - 对称双向块匹配



# 运动补偿去隔行 (cont'd)

- 运动矢量的反向扩展

- 将搜索扩展到前面两场，以尽量减少空域子像素插值



使用零阶保持滤波扩展  
到前两场

- 时间递归去隔行

- 使用了前面的去隔行帧

# 帧/场速率转换

---

- 运动图像到 TV
  - 24 Hz 到 50i
  - 24 Hz 到 60i
- 50i 到 60i
- 运动图像模式检测
- 场速率转换: 50i/60i 到 100i/120i

# 24Hz到 50i/60i 的转换

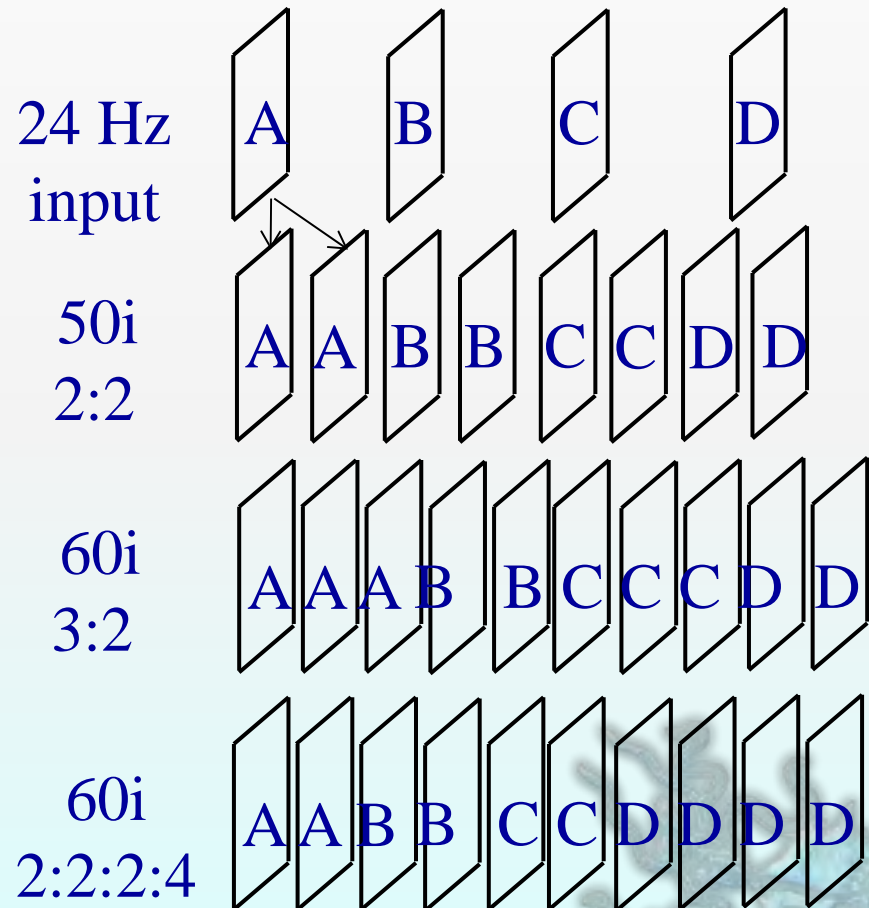
- 输入 24 Hz 逐行形式

- 输出 50i

- 2:2 偶/奇分离 48i
- 电影，稍快于 50i TV

- 输出 60i

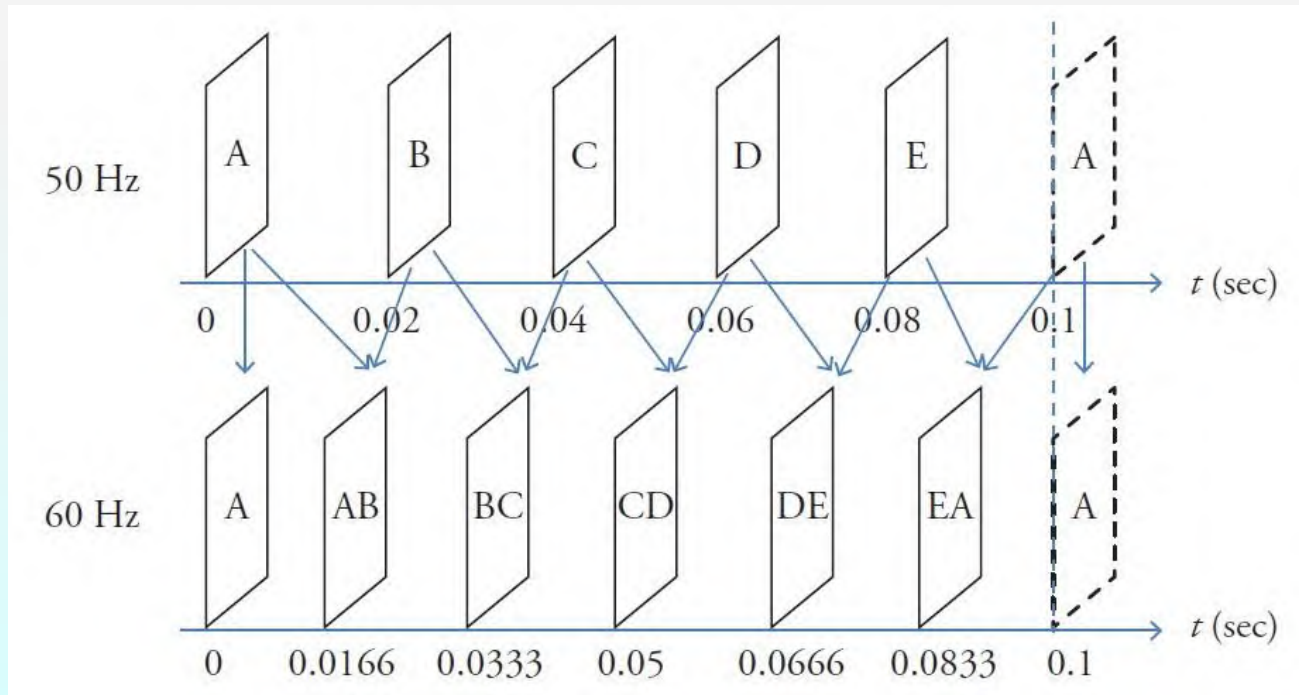
- 3: 2下拉方式
- 2: 2: 2: 4下拉方式





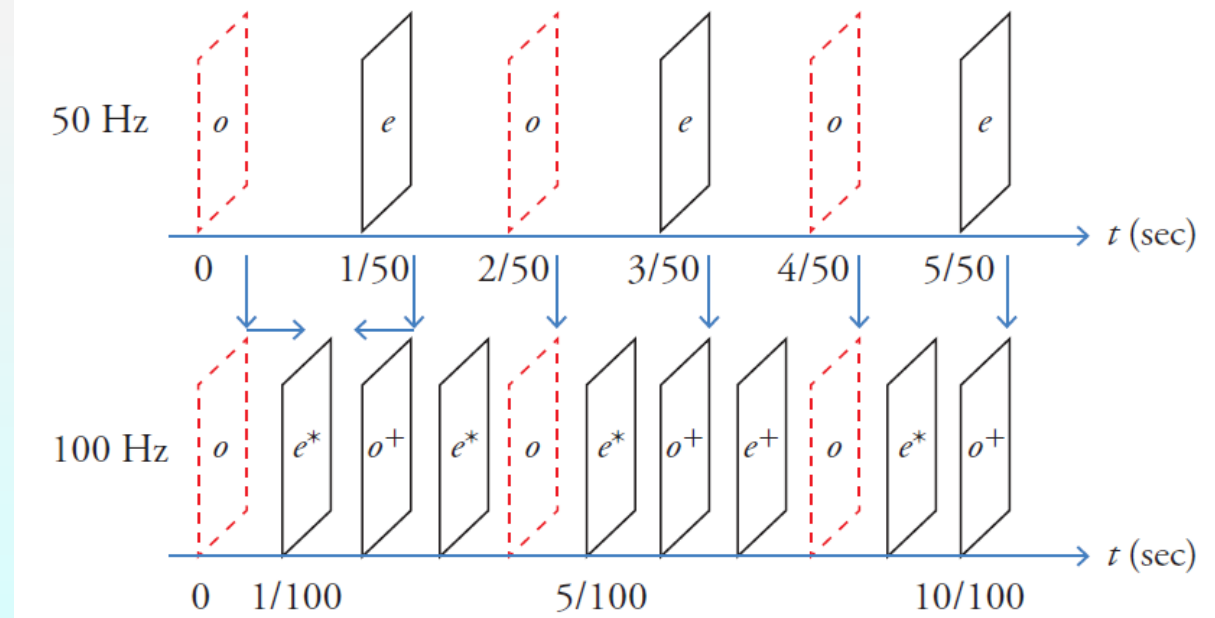
# 50i 到 60i 的转换

- **50i 至 60i** - 需要每五张照片复制一张图像（偶数和奇数场）。
- **60i 至 50i** 转换可以通过每六帧丢弃一幅图像来实现。
- 可以通过**时域加权内插**（混合）获得更平滑的结果。



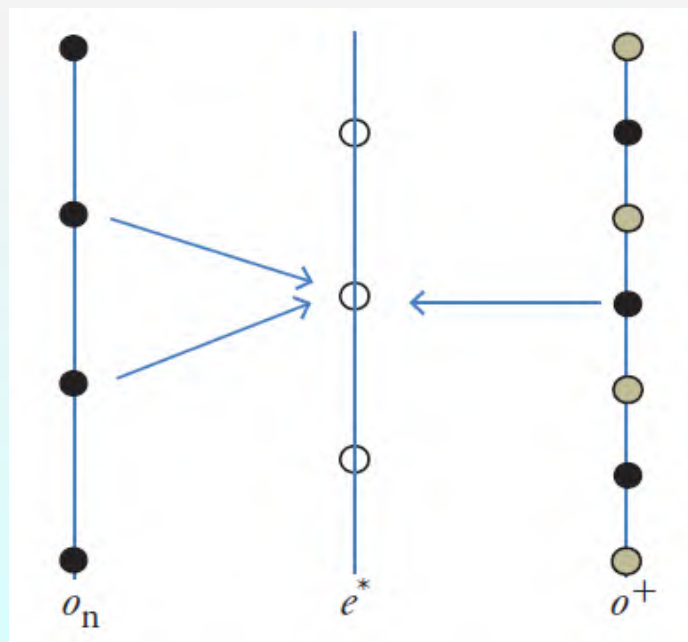
# 场速率转换: 50i/60i 到 100i/120i

- **场复制**: 重复奇数场以形成下一个偶数场, 重复偶数场以形成下一个奇数场。这在移动区域具有相当好的表现, 但在静止区域表现不佳。
- 为了**保留场的时间排序**, 只有输入奇数场 (标记为 $o$ ) 在100 Hz输出序列中保持“原样”。通过场内行平均将输入偶数场转换为奇数场 (标记为 $o^+$ )。



## 场速率转换 (cont'd)

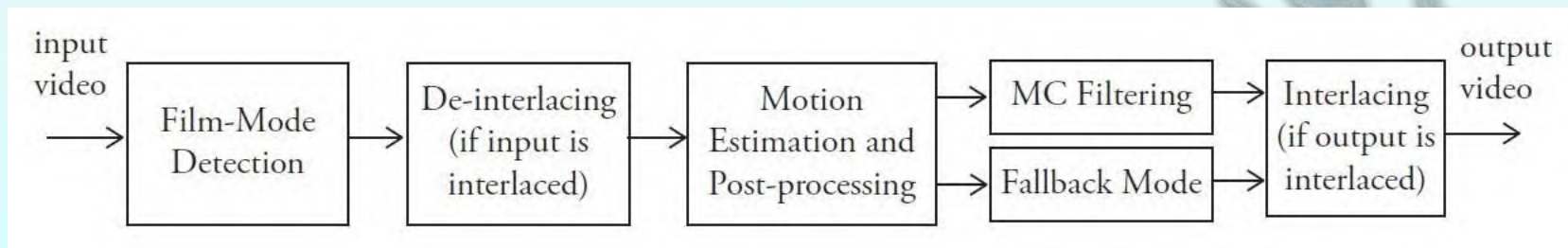
- 通过**场间过滤（混合）**，在 $o$ 和 $o^+$ 之间插入**新的偶数场**（标记为 $e^*$ ）。图6.18中箭头所示的**三点平均**或**中值滤波器**用于偶数场的插值。三点过滤器使用输入奇数和偶数场中的现有行。



**黑色圈**表示输入奇数和偶数场中存在的线。**浅色圆圈**表示通过帧内行平均从输入偶数场插值出的输出奇数场行。**偶数场 $e^*$** 由三点滤波器插值。在场 $o^+$ 中由**黑圈**表示的偶数行被丢弃。

# 运动补偿帧/场速率转换

- 在**空间图像细节准确性**（重用原始帧）和**时空准确性**（在给定时刻根据运动轨迹得到的帧内对象位置）之间存在折中问题。如果我们在与其实际归属稍有不同的时间重用现有帧，则空间细节将被很好地保留，但位置不准确可能导致**运动抖动**。如果我们为准确的时间位置合成一个新的帧，那么对象的空间位置将是正确的（假设是真实的运动估计），但由于插值误差，某些**空间细节可能会被模糊**。
- 如果输入视频是**隔行扫描**，则在运动补偿滤波之前应用**去隔行扫描**，以便即使期望输出为隔行扫描也能进行更为准确的运动估计和帧渲染。





# 运动自适应噪声滤波

- **运动自适应滤波**是对应于边缘保持空间滤波的时域形式，因为帧与帧之间的运动会产生时间边缘。
- **隐式运动自适应滤波器**：是用于边缘保持空间滤波器结构的扩展，包括方向滤波器和顺序统计滤波器，如中值，加权中值和多级中值滤波器。
- **基于运动检测的滤波器**可以是FIR或IIR滤波器。
- 运动自适应FIR滤波器的**简单示例**可由下式给出

$$\hat{s}(n_1, n_2, k) = (1-\gamma)g(n_1, n_2, k) + \gamma g(n_1, n_2, k-1)$$

而其**IIR滤波器**为：

$$\hat{s}(n_1, n_2, k) = (1-\gamma)g(n_1, n_2, k) + \gamma \hat{s}(n_1, n_2, k-1)$$

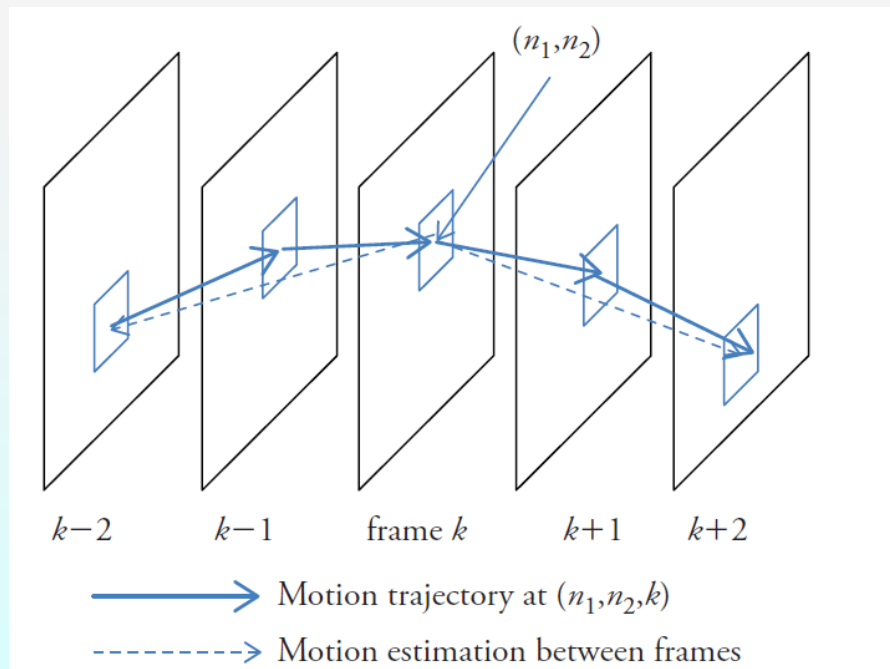
其中： $\gamma = \max\{0, 1/2 - \alpha |g[n_1, n_2, k] - g[n_1, n_2, k-1]|\}$

为运动检测信号， $\alpha$ 为尺度常数。

# 运动补偿噪声滤波

- 沿每个像素的运动轨迹去噪

- 当运动估计准确时，滤波应接近直接平均。
- 如果运动估计不准确，则应关闭滤波或进行加权平均，以防止运动区域出现模糊。



- MC-LMMSE 滤波器

- MC自适应加权平均 (MC-AWA) 滤波器

运动估计

- 子像素精度
- 对于遮挡和噪声应足够鲁棒

# MC-LMMSE 滤波器

- 在像素  $(n_1, n_2, k)$  处的估计为：

$$\hat{s}[n_1, n_2, k] = \mu_s[n_1, n_2, k] + \frac{\sigma_s^2(n_1, n_2, k)}{\sigma_s^2(n_1, n_2, k) + \sigma_v^2} (g[n_1, n_2, k] - \mu_g[n_1, n_2, k])$$

其中  $\mu_s(n_1, n_2, k)$  和  $\sigma_s^2(n_1, n_2, k)$  表示在有噪视频中像素  $(n_1, n_2, k)$  在其运动轨迹上的样本均值和方差：

$$\hat{\mu}_g(n_1, n_2, k) = \frac{1}{L} \sum_{(i_1, i_2, l) \in \mathbf{S}_{n_1, n_2, k}} g[i_1, i_2, l]$$

$$\hat{\sigma}_g^2(n_1, n_2, k) = \frac{1}{L} \sum_{(i_1, i_2, l) \in \mathbf{S}_{n_1, n_2, k}} (g[i_1, i_2, l] - \mu_g[n_1, n_2, k])^2$$

而  $L$  为  $\mathbf{S}_{n_1, n_2, k}$  中的像素数，则：

$$\hat{\sigma}_s^2(n_1, n_2, k) = \max\{\hat{\sigma}_g^2(n_1, n_2, k) - \hat{\sigma}_v^2, 0\}$$

# MC-LMMSE滤波器的适应性

- 假定:

- 噪声为零均值和白色的。
- 残差  $\rho(m, n; k) = s(m, n; k) - \mu_s(m, n; k)$  是白色的, 且具有非稳态方差  $\sigma_s^2(m, n; k)$ 。
- 在每个像素处沿着运动轨迹使用样本均值和方差以估计 **整体均值和方差**。

- 鲁棒运动估计:

- 如果运动估计 **性能良好**:

$$\hat{\sigma}_s^2(\mathbf{n}, k) \approx 0 \Rightarrow \hat{s}(\mathbf{n}) \approx \hat{\mu}_g(\mathbf{n}, k)$$

- 如果运动估计 **性能不好**:

$$\hat{\sigma}_s^2(\mathbf{n}, k) \gg \hat{\sigma}_v^2(\mathbf{n}, k) \Rightarrow \hat{s}(\mathbf{n}) \approx g(\mathbf{n}, k)$$

- 当场景 **内容突然变化** (例如, 快速变焦, 相机镜头和/或照明的变化) 时, 大多数噪声滤波器表现不佳。



# MC-AWA 滤波器

- MC-AWA是可用于视频去噪的**双边滤波器的扩展**。
- 像素 $(n_1, n_2, k)$ 的估计**由下式给出：

$$s(n_1, n_2; k) = \sum_{(i_1, i_2; l) \in T(n_1, n_2; k)} w(i_1, i_2; l) g(i_1, i_2; l)$$

其中 $T(n_1, n_2, k)$  表示**运动轨迹**

$$w(i_1, i_2; l) = \frac{K(n_1, n_2; k)}{1 + a \max\{\varepsilon^2, [g(n_1, n_2; k) - g(i_1, i_2; l)]^2\}}$$

为权重，

$$K(n_1, n_2; k) = \frac{1}{\sum_{(i_1, i_2; l) \in T(n_1, n_2; k)} \frac{1}{1 + a \max\{\varepsilon^2, [g(n_1, n_2; k) - g(i_1, i_2; l)]^2\}}}$$

是归一化常数， $a$ 是惩罚参数， $\varepsilon^2 = 2\sigma_{\text{noise}}^2$

## MC-AWA滤波器的自适应性 (cont'd.)

- 阈值参数 $\varepsilon$  通常选择为 $\varepsilon = 2\sigma_v^2$
- 在当前像素 $g(n_1, n_2; k)$  与沿着运动轨迹的所有其他像素之间的差值小于 $\varepsilon$ 时，所有权重都达到值 $1 / (\text{沿着轨迹的}\# \text{个帧})$ ，此时AWA降为直接平均。
- 当 $g(n_1, n_2; k)$  与轨迹上特定像素的强度（如 $g(i_1, i_2; l)$ ）大于 $\varepsilon$ 时， $g(i_1, i_2; l)$  的贡献按下列因子加权：

$$w(i_1, i_2; l) = \frac{K(n_1, n_2; k)}{1 + a \max\left\{\varepsilon^2, [g(n_1, n_2; k) - g(i_1, i_2; l)]^2\right\}}$$

- MC-AWA滤波器对于运动估计误差和突发场景变化是鲁棒的。

# BM4D 滤波器

- **BM4D** 通过利用变换域中**非局部空间**、**时间自相似性**和**稀疏性**，将BM3D框架中性能提升较强的协同滤波能力进行扩展，即从图像去噪扩展到**视频滤波**。
- 即使在存在快速运动的情况下，沿着**运动轨迹的块的相似性**比**单帧内的非局部相似性**更强。
- 一个更早的将BM3D扩展到视频去噪的方法被称为**V-BM3D**，是将从一组连续帧提取的类似块分组为3D阵列，而不管它们是来自时间相似性还是非本地空间相似性。
- 与之不同，V-BM4D将视频信号分成**相互近似的时空体组**，即一组根据非局部搜索过程计算的特定轨迹的视频块序列形成的3D结构。因此，V-BM4D中的组是**3D立体的4D数组**，最后通过可分离的4D时空变换进行协同滤波。

# 多帧复原

- 矢量矩阵模型可以扩展到**L帧的多帧建模**:

$$\mathbf{g} = \mathbf{D} \mathbf{s} + \mathbf{v}$$

其中:

$$\mathbf{g} = \begin{bmatrix} \mathbf{g}_1 \\ \vdots \\ \mathbf{g}_L \end{bmatrix}, \quad \mathbf{s} = \begin{bmatrix} \mathbf{s}_1 \\ \vdots \\ \mathbf{s}_L \end{bmatrix}, \quad \mathbf{v} = \begin{bmatrix} \mathbf{v}_1 \\ \vdots \\ \mathbf{v}_L \end{bmatrix}$$

为 $N^2L \times 1$ 向量, 分别表示作为多帧向量的**观测图像**, **理想图像**和**噪声图像**的矢量,

$$\mathbf{D} = \begin{bmatrix} \mathbf{D}_1 & \cdots & \mathbf{0} \\ \vdots & \ddots & \vdots \\ \mathbf{0} & \cdots & \mathbf{D}_L \end{bmatrix}$$

是一个表示多帧模糊算子的 $N^2L \times N^2L$  矩阵。

- **多帧模糊矩阵D**是块对角阵, 表示没有时间模糊。



## 多帧复原 (cont'd)

- **L 帧的多帧维纳估计  $\hat{\mathbf{s}}$  为：**

$$\hat{\mathbf{s}} = (\mathbf{D}^T \mathbf{D} + \mathbf{R}_s^{-1} \mathbf{R}_v)^{-1} \mathbf{D}^T \mathbf{g}$$

其中：

$$\hat{\mathbf{s}} = \begin{bmatrix} \hat{s}_1 \\ \vdots \\ \hat{s}_L \end{bmatrix} \quad \mathbf{R}_s = \begin{bmatrix} \mathbf{R}_{s;11} & \cdots & \mathbf{R}_{s;1L} \\ \vdots & \ddots & \vdots \\ \mathbf{R}_{s;L1} & \cdots & \mathbf{R}_{s;LL} \end{bmatrix} \quad \mathbf{R}_v = \begin{bmatrix} \mathbf{R}_{v;11} & \cdots & \mathbf{R}_{v;1L} \\ \vdots & \ddots & \vdots \\ \mathbf{R}_{v;L1} & \cdots & \mathbf{R}_{v;LL} \end{bmatrix}$$

$$\mathbf{R}_{s;ij} = \mathbf{E}\{\mathbf{s}_i \mathbf{s}_j^T\} \quad \mathbf{R}_{v;ij} = \mathbf{E}\{\mathbf{v}_i \mathbf{v}_j^T\}, i, j=1, 2, \dots, L.$$

- 如果  $\mathbf{R}_{s;ij} = \mathbf{0}$  对  $i \neq j, i, j=1, 2, \dots, L$ , 则多帧估计就相当于将多个单独获得的L个单帧估计组合。
- 两种有效实现方法：
  - 互相关多维维纳滤波器 (CCMF)
  - 运动补偿多帧维纳滤波器 (MCMF)