

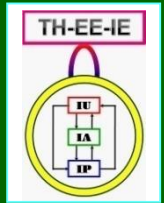
图象工程（上）

图 象 处 理

（第4版）

章毓晋

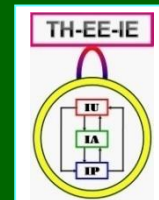
清华大学电子工程系 100084 北京



第4单元 拓展技术

- 第12章 图象水印
- 第13章 彩色图象处理
- 第14章 视频图象处理
- 第15章 多尺度图象处理

对图象的研究和应用一直是一个活跃的领域，新的理论、新的方法不断涌现，新的技术、新的手段也在不断拓展。基于前三个单元介绍的图象处理技术，进一步拓展

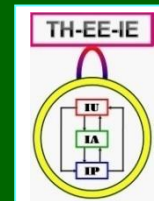


第14章 视频图象处理

视频一般代表一类彩色序列图象，它描述了在一段时间内3-D景物投影到2-D图象平面且由3个分离的传感器获得的场景辐射强度

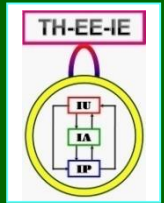
视频可看作是对（静止）图象的扩展。除了原来图象的一些概念和定义仍然保留外，为表示视频还需要一些新的概念和定义

视频相对图象最明显的一个区别就是含有场景中的随时间运动变化的信息，这也是使用视频的一个主要目的。针对含有运动信息的视频的特点，原来的图象处理技术也需要相应的推广



第14章 视频图象处理

- 14.1 视频表达和格式
- 14.2 运动分类和表达
- 14.3 运动检测
- 14.4 视频滤波
- 14.5 视频预测编码

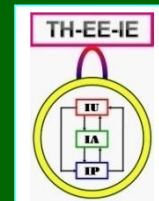


14.1 视频表达和格式

{要讨论视频图象处理，首先要讨论视频的表示或表达，以及视频的格式和显示等}

14.1.1 视频基础

14.1.2 彩色电视制式



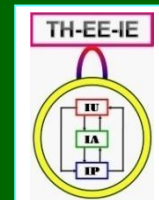
14.1.1 视频基础

1. 视频表达函数

函数 $f(x, y, t)$ 描述了在时间 t 投影到图象平面 XY 的3-D景物的某种性质

对彩色图象用函数 $f(x, y)$ 来表示，考虑视频从灰度到彩色的扩展，视频可用函数 $f(x, y, t)$ 来表示

彩色视频可由3个函数（它们组成一个矢量函数）来表示，每个函数描述一个彩色分量。这种格式的视频称之为分量视频。复合视频格式中的3个彩色信号被复用成一个单独的信号



14.1.1 视频基础

2. 视频彩色模型

$YC_B C_R$ 模型: Y 代表亮度分量

$$Y = rR + gG + bB$$

色度分量 C_B 表示蓝色部分与亮度值的差, 而色度分量 C_R 表示红色部分与亮度值的差

$$\begin{aligned} C_B &= B - Y \\ C_R &= R - Y \end{aligned}$$

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1.0 & -0.00001 & 1.40200 \\ 1.0 & -0.34413 & -0.71414 \\ 1.0 & 1.77200 & 0.00004 \end{bmatrix} \begin{bmatrix} Y \\ C_B \\ C_R \end{bmatrix}$$

14.1.1 视频基础

3. 视频空间采样率

⇒ 对亮度分量 Y 的采样率

4 : 4 : 4格式，即对亮度分量 Y 的采样率与对色度分量 C_B 和 C_R 的采样率相同

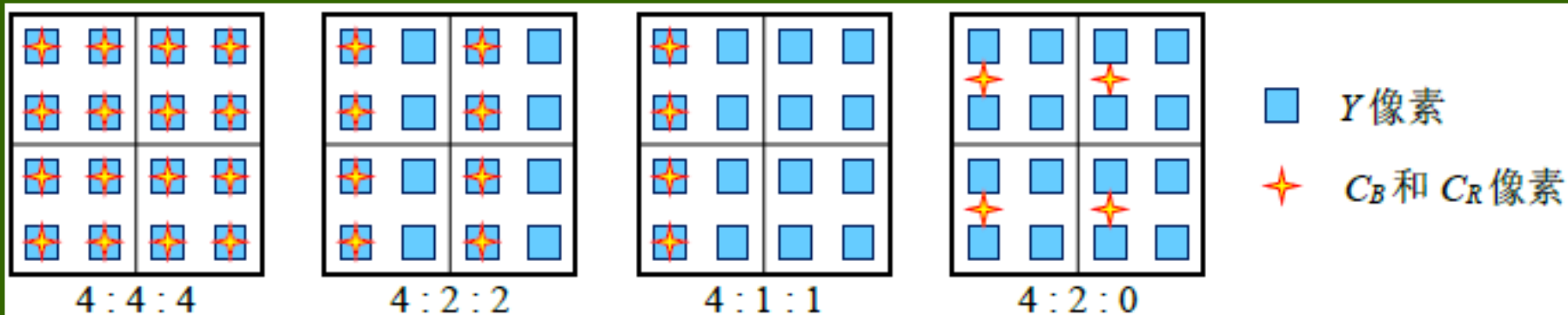
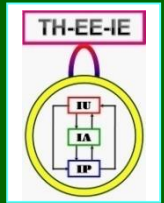


图 14.1.2 4种采样格式（两个相邻行属于两个不同的场）示例



14.1.1 视频基础

4. 视频显示

显示器的宽高比主要有4 : 3和16 : 9两种
逐行扫描以帧为单位

隔行扫描以场为单位（一帧分为两场：顶场包含所有奇数行，底场包含所有偶数行）

视频在显示时还需要有一定的帧率，即相邻两帧出现的频率。根据人眼的视觉暂留特性，帧率需要高于25帧/秒，低了会出现闪烁和不连续

14.1.1 视频基础

5. 视频码率

视频的数据量由视频的时间分辨率、空间分辨率和幅度分辨率共同决定 (帧率)

存储一秒钟视频图象所需的位数 b (也称为视频码率, 单位是b/s)

$$b = L \times M \times N \times k$$

(对照(1.1.9)式)

视频的数据量也可由行数 f_y 、每行样本数 f_x 和帧频 f_t 来定义

$$b = f_x f_y f_t K$$

K : 象素比特数

14.1.2 彩色电视制式

YUV模型

在PAL制和SECAM制系统中使用

Y 代表亮度分量, U 和 V 分别正比于色差 $B - Y$ 和 $R - Y$, 称为色度分量

$$\begin{bmatrix} Y \\ U \\ V \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.147 & -0.289 & 0.436 \\ 0.615 & -0.515 & -0.100 \end{bmatrix} \begin{bmatrix} R' \\ G' \\ B' \end{bmatrix}$$

经过伽
马校正

$$\begin{bmatrix} R' \\ G' \\ B' \end{bmatrix} = \begin{bmatrix} 1.000 & 0.000 & 1.140 \\ 1.000 & -0.395 & -0.581 \\ 1.000 & 2.032 & 0.001 \end{bmatrix} \begin{bmatrix} Y \\ U \\ V \end{bmatrix}$$

14.1.2 彩色电视制式

YIQ模型

在NTSC制系统中使用

Y 代表亮度分量, I 和 Q 分别是 (PAL制) U 和 V 分量旋转 33° 后的结果

$$\begin{bmatrix} Y \\ I \\ Q \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ 0.596 & -0.275 & -0.321 \\ 0.212 & -0.523 & 0.311 \end{bmatrix} \begin{bmatrix} R' \\ G' \\ B' \end{bmatrix}$$

经过伽
马校正

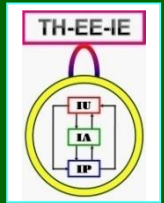
$$\begin{bmatrix} R' \\ G' \\ B' \end{bmatrix} = \begin{bmatrix} 1.000 & 0.956 & 0.620 \\ 1.000 & -0.272 & -0.647 \\ 1.000 & -1.108 & 1.700 \end{bmatrix} \begin{bmatrix} Y \\ I \\ Q \end{bmatrix}$$

14.1.2 彩色电视制式

由于人眼对色度信号的分辨能力较低，所以在普通电视制式中均对色度信号采用比对亮度信号更低的空间采样率以降低视频数据量

表 14.1.3 普通电视制式的空间采样率

电视制式	亮度分量		色度分量		$Y:U:V$
	行数	像素/行	行数	像素/行	
NTSC	480	720	240	360	4:2:2
PAL	576	720	288	360	4:2:2
SECAM	576	720	288	360	4:2:2



14.2 运动分类和表达

1. 运动分类

前景运动:

目标在场景中的自身运动, 又称为局部运动

背景运动:

主要由摄像机的运动所造成, 帧图象内所有点整体移动, 又称为全局运动或摄像机运动。一般具有整体性强、比较规律的特点

前景和背景的运动或静止共有4种组合{P.331}

14.2 运动分类和表达

1. 运动分类

摄像机运动

- 沿轴运动(3)
- 绕轴运动(3)
- 变焦运动(1)
(放大/缩小)

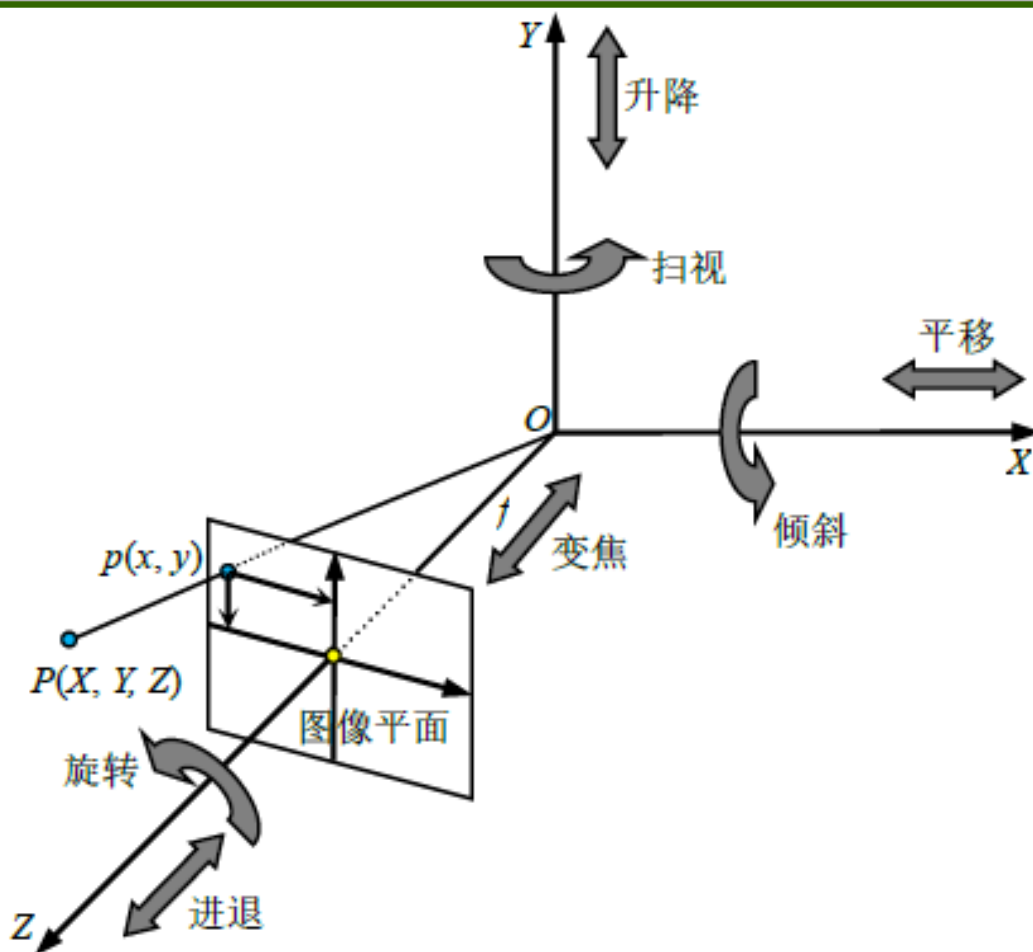


图 14.2.1 摄像机的运动

14.2 运动分类和表达

2. 运动矢量场表达

- 将整幅图象分成许多固定大小的块
- 用矢量表示块的运动
- 将每个运动矢量用（有起点）无箭头的线段（线段长度与矢量大小亦即运动速度成正比）来表示，并叠加在原始图象上



图 14.2.2 全局运动矢量叠加在原图上

14.2 运动分类和表达

3. 运动直方图表达

- 运动矢量方向直方图
仅保留运动方向信息以减少数据量
- 运动区域类型直方图
统计各个运动区域满足不同运动模型的像素

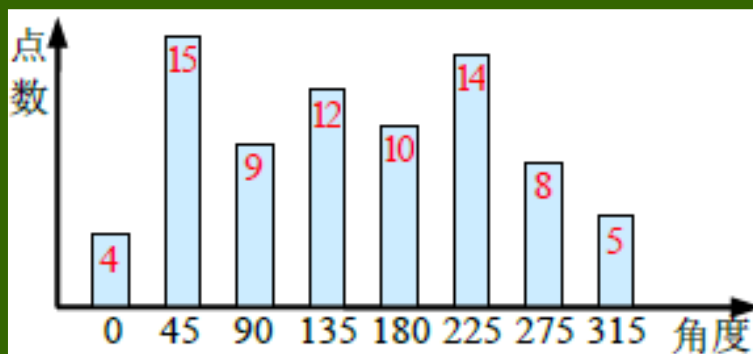


图 14.2.3 运动矢量方向直方图

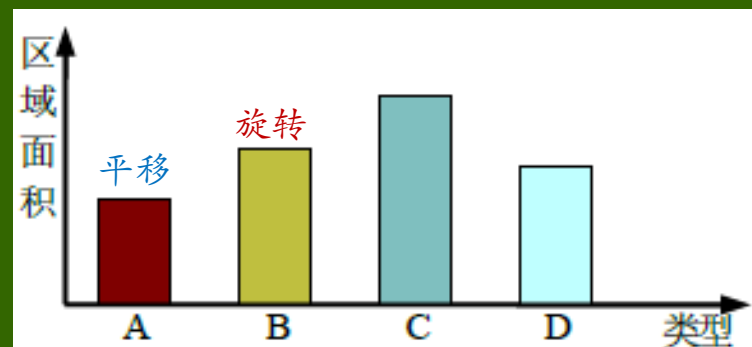
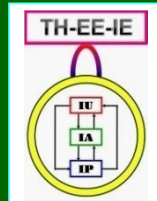


图 14.2.4 运动区域类型直方图



14.2 运动分类和表达

4. 运动轨迹表达

运动轨迹由一系列关键点以及一组在这些关键点之间进行插值的函数来描述

$$f(t) = f_p(t) + v_p(t - t_p) + a_p(t - t_p)^2 / 2$$

p : 时间轴一点, v_p : 运动速度, a_p : 运动加速度

零次函数: $x(t) = x(t_0)$

一次函数: $x(t) = x(t_1) + v(t_1) \cdot (t - t_1)$

两次函数: $x(t) = x(t_2) + v(t_2) \cdot (t - t_2) + a(t_2) \cdot (t - t_2)^2 / 2$

14.2 运动分类和表达

4. 运动轨迹表达

运动轨迹表达了目标在运动过程中的位置信息
国际标准MPEG-7推荐了一个运动轨迹描述符

$$f(t) = f_p(t) + v_p(t - t_p) + a_p(t - t_p)^2 / 2$$

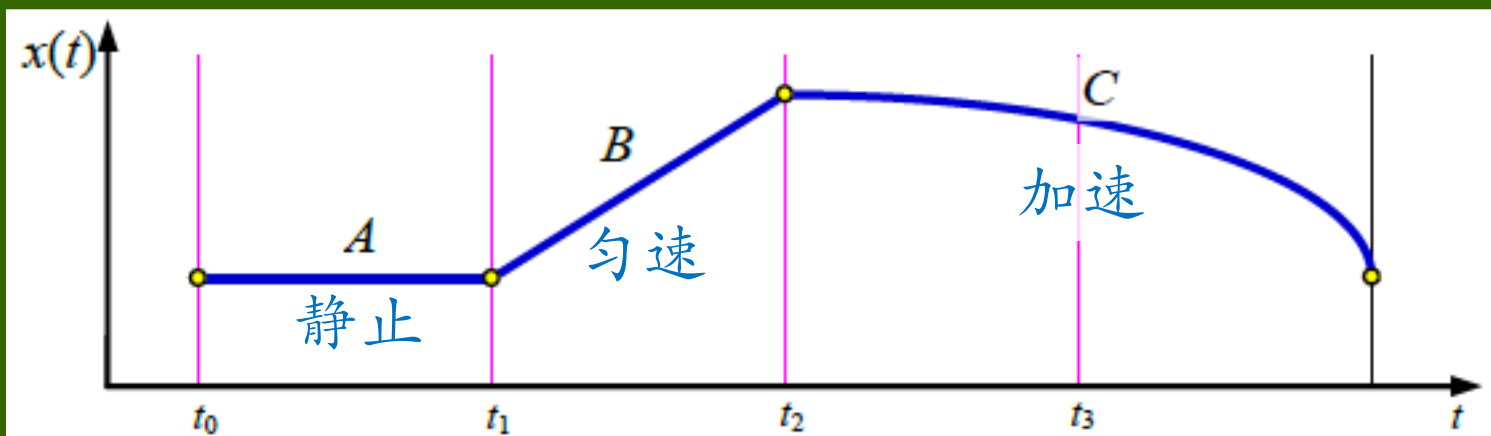
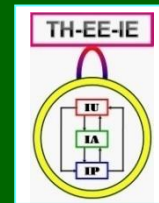


图 14.2.5 轨迹描述中关键点和插值函数示意图



14.3 运动检测

{ ① 哪些景物有运动, ② 运动的情况如何 }

14.3.1 利用图象差的运动检测

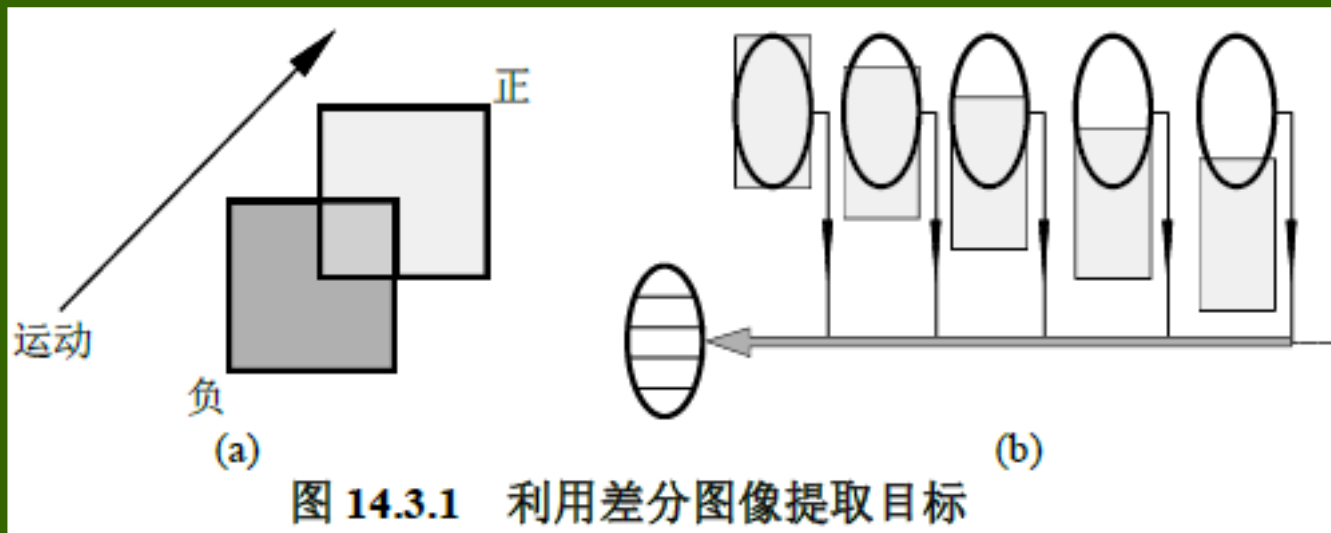
14.3.2 基于模型的运动检测

14.3.3 频率域运动检测

14.3.1 利用图象差的运动检测

1. 差图象的计算

- 通过逐像素比较可直接求取前后两帧图象之间的差别
- 差图象不为零处表明该处的像素发生了移动



算术运算
逻辑运算

14.3.1 利用图象差的运动检测

1. 差图象的计算

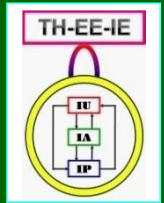
设在两个时刻 t_i 和 t_j 采集到两幅图象 $f(x, y, t_i)$ 和 $f(x, y, t_j)$, 则据此可得到差图象 (模板) :

$$d_{ij}(x, y) = \begin{cases} 1 & \text{如 } |f(x, y, t_i) - f(x, y, t_j)| > T_g \\ 0 & \text{其他} \end{cases}$$

灰度阈值

? 差图象为1时的不同情况:

如: $f(x, y, t_i)$ 是一个运动目标的象素而
 $f(x, y, t_j)$ 是一个背景象素、或是同一个
 运动目标但不同位置的象素,



14.3.1 利用图象差的运动检测

1. 差图象的计算

灰度差异显著性的判别：

似然比

$$\frac{\left[\frac{\sigma_i + \sigma_j}{2} + \left(\frac{\mu_i - \mu_j}{2} \right)^2 \right]^2}{\sigma_i \bullet \sigma_j} > T_s$$

各 μ 和 σ 分别是在时刻 t_i 和 t_j 采集到的两幅图象上对应观测窗口中的均值和方差， T_s 是显著性阈值

14.3.1 利用图象差的运动检测

2. 累积差图象的计算

累积差图象ADI (Accumulative Difference Image)

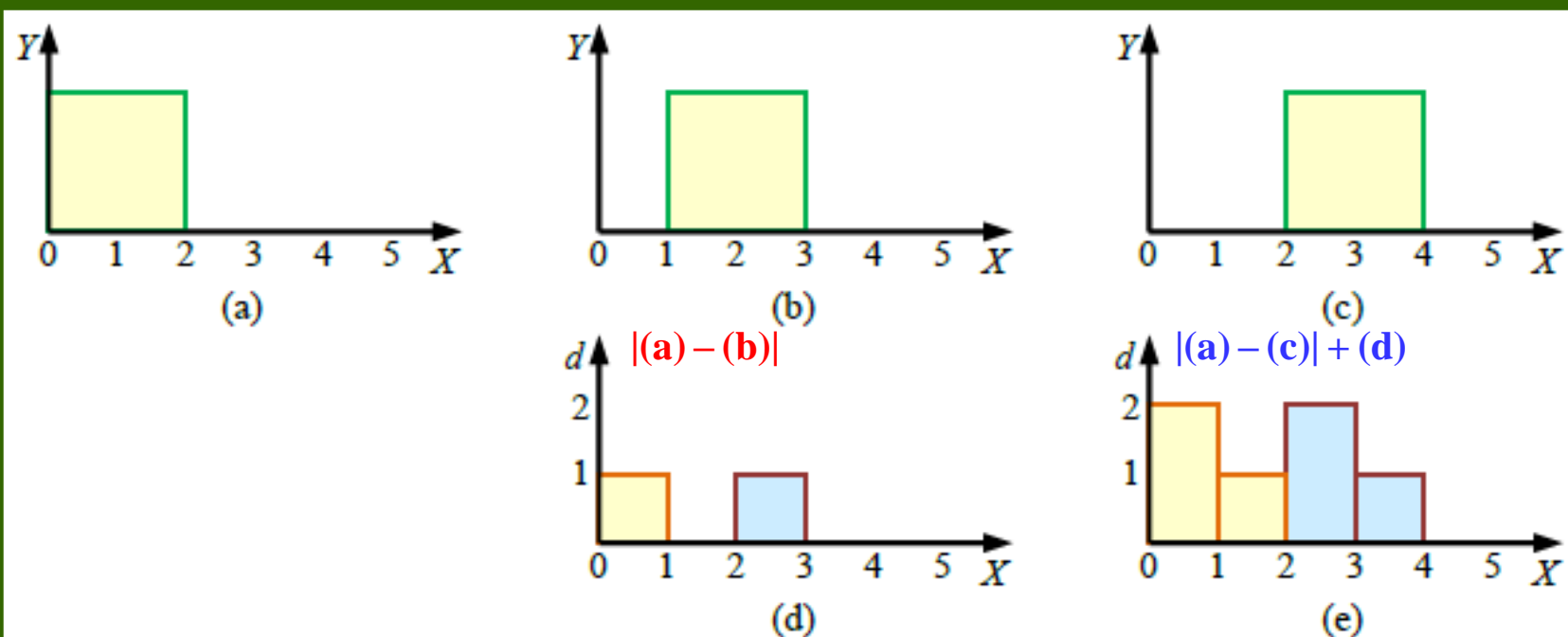
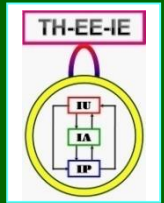


图 14.3.2 利用累积差图像确定目标移动

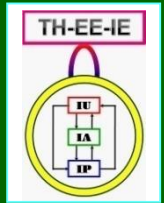


14.3.1 利用图象差的运动检测

2. 累积差图象的计算

ADI有三个功能：

- (1) ADI中相邻像素值间的梯度关系可用来估计目标移动的速度矢量，这里梯度的方向就是速度的方向，梯度的大小与速度成正比
- (2) ADI中像素的个数（值）可帮助确定运动目标的尺寸和移动的距离
- (3) ADI中包含了目标运动的全部历史资料，有助于检测慢运动和尺寸较小目标的运动



14.3.1 利用图象差的运动检测

2. 累积差图象的计算

三种ADI图象:

(1) 绝对

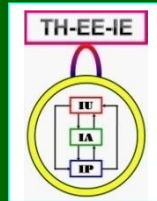
$$A_k(x, y) = \begin{cases} A_{k-1}(x, y) + 1 & \text{如果 } |f(x, y, t_1) - f(x, y, t_k)| > T_g \\ A_{k-1}(x, y) & \text{其他} \end{cases}$$

(2) 正

$$P_k(x, y) = \begin{cases} P_{k-1}(x, y) + 1 & \text{如果 } [f(x, y, t_1) - f(x, y, t_k)] > T_g \\ P_{k-1}(x, y) & \text{其他} \end{cases}$$

(3) 负

$$N_k(x, y) = \begin{cases} N_{k-1}(x, y) + 1 & \text{如果 } [f(x, y, t_1) - f(x, y, t_k)] < -T_g \\ N_{k-1}(x, y) & \text{其他} \end{cases}$$

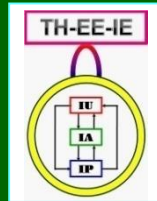


14.3.1 利用图象差的运动检测

2. 累积差图象的计算

由ADI图象获得的信息：

- (1) 正ADI图象中的非零区域面积等于运动目标的面积
- (2) 正ADI图象中对应运动目标的位置也就是运动目标在参考图中的位置
- (3) 当正ADI图象中的运动目标移动到与参考图中的运动目标不重合时，正ADI图象停止计数
- (4) 绝对ADI图象包含了正ADI图象和负ADI图象中的所有目标区域
- (5) 运动目标的运动方向和速度可根据绝对ADI图象和负ADI图象来确定



14.3.2 基于模型的运动检测

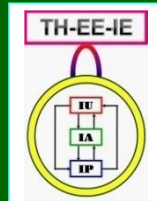
运动模型

考虑对摄像机建模来进行全局运动检测

全局运动通用模型
$$\begin{cases} u = f_u(x, y, k_0, k_1, k_2, \dots) \\ v = f_v(x, y, k_0, k_1, k_2, \dots) \end{cases}$$

6参数仿射模型
$$\begin{cases} u = k_0 x + k_1 y + k_2 \\ v = k_3 x + k_4 y + k_5 \end{cases}$$

8参数双线性模型
$$\begin{cases} u = k_0 xy + k_1 x + k_2 y + k_3 \\ v = k_4 xy + k_5 x + k_6 y + k_7 \end{cases}$$



14.3.3 频率域运动检测

借助傅里叶变换把检测转到频率域

(1) 对平移的检测

$$f(x + dx, y + dy, t_{k+1}) = f(x, y, t_k)$$

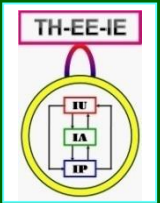
根据傅里叶变换

$$F_k(u, v) \Leftrightarrow f(x, y, t_k)$$

$$F_{k+1}(u, v) \Leftrightarrow f(x + dx, y + dy, t_{k+1})$$

借助平移性质

$$F_{k+1}(u, v) = F_k(u, v) \exp[\underbrace{j2\pi(udx + vdy)}_{\text{相位角之差}}]$$



14.3.3 频率域运动检测

(1) 对平移的检测

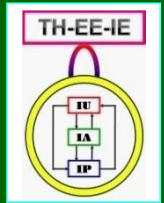
相位角之差

$$d\theta(u, v) = 2\pi(udx + vdy)$$

考虑到傅里叶变换的分离性

$$dx = \frac{d\theta_x(u)}{2\pi u} \quad dy = \frac{d\theta_y(v)}{2\pi v}$$

$d\theta_x(u)$ 和 $d\theta_y(v)$ 分别为 $f(x, y, t_k)$ 和 $f(x, y, t_{k+1})$ 在 X 轴上和 Y 轴上投影的傅里叶变换的相位角之差



14.3.3 频率域运动检测

(2) 对旋转的检测

借助傅里叶变换功率谱进行

图象中的直线模式在傅里叶功率谱中对应过频谱原点的直线模式

$$P_k(u, v) = |F_k(u, v)|^2 \quad P_{k+1}(u, v) = |F_{k+1}(u, v)|^2$$

在 $P_k(u, v)$ 和 $P_{k+1}(u, v)$ 中分别搜索相对应的、过原点的直线模式，再计算直线间的夹角得到旋转角

14.3.3 频率域运动检测

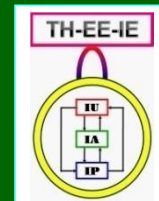
(3) 对尺度变化的检测

图象空间的尺度变化对应傅里叶变换域中频率高低的变化

- 1) 在 $P_k(u, v)$ 和 $P_{k+1}(u, v)$ 中分别搜索方向相同的直线模式 L_k 和 L_{k+1}
- 2) 将 L_k 投影到 $P_{k+1}(u, v)$ 上, 得到 L'_k

$$S = \frac{|L'_k|}{|L_{k+1}|} \quad \text{"|•|"对应长度}$$

如果 $S < 1$, 表明目标图象尺寸增加了 S 倍



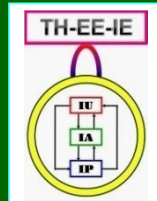
14.4 视频滤波

滤波在这里代表多种视频处理过程和手段
分为空域滤波器（仅考虑帧内）和时空滤波器（还考虑帧间）两类

14.4.1 基于运动检测的滤波

14.4.2 基于运动补偿的滤波

14.4.3 消除匀速直线运动模糊



14.4.1 基于运动检测的滤波

1. 直接滤波

➤ 帧平均技术

不影响帧图象空间分辨率的情况下消除噪声

沿时间轴的1-D滤波，即进行时域平均，在场景中有突然的随时间变化处会导致时域模糊

运动适应滤波，它利用相邻帧间的运动信息来确定滤波方向，将由于运动造成的沿时间轴变化与噪声导致的变化区别开来

14.4.1 基于运动检测的滤波

2. 利用运动检测信息

➤ 有限脉冲响应 (FIR) 滤波器

$$\hat{f}(x, y, t) = (1 - \beta)g(x, y, t) + \beta g(x, y, t - 1)$$

仅输入图象

具有有限的噪声消除能力

➤ 无限脉冲响应 (IIR) 滤波器

$$\hat{f}(x, y, t) = (1 - \beta)g(x, y, t) + \beta \hat{f}(x, y, t - 1)$$

上一次结果

有更强的噪声消除能力 (但相位可能失真)

$$\beta = \max \left\{ 0, \frac{1}{2} - \alpha |g(x, y, t) - g(x, y, t - 1)| \right\}$$

{P.340}

14.4.2 基于运动补偿的滤波

像素灰度在确定的运动轨迹上保持不变

1. 运动轨迹和时空频谱

运动轨迹 (坐标)

$$M(t'; x, y, t_0) = (x', y')$$

速度

$$s(x', y', t') = \left. \frac{dM}{dt}(t; x, y, t_0) \right|_{t=t'}$$

帧间灰度变化

$$\begin{aligned} f_M(x, y, t) &= f_M(x - s_x t, y - s_y t, 0) \\ &\approx f_0(x - s_x t, y - s_y t) \end{aligned}$$

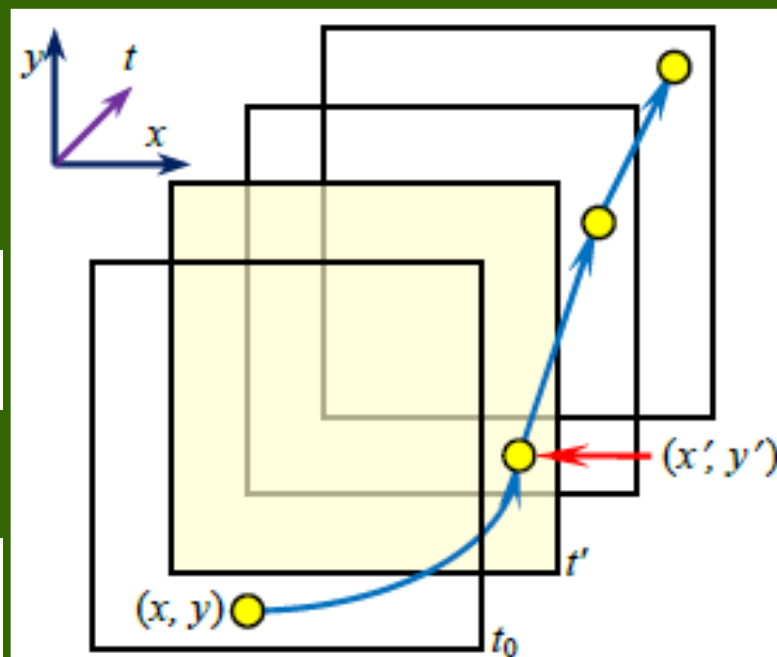


图 14.4.1 运动轨迹

14.4.2 基于运动补偿的滤波

1. 运动轨迹和时空频谱

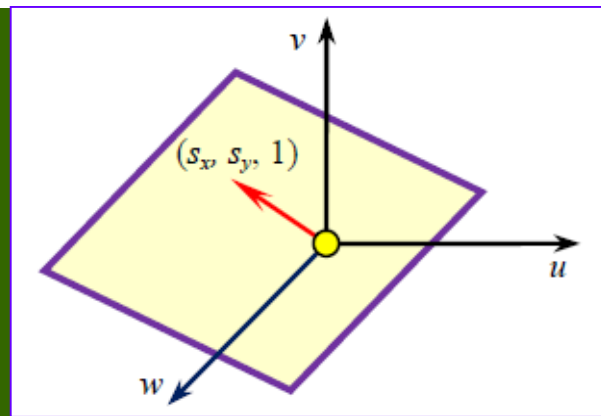
时空函数的傅里叶变换

$$F_M(u, v, w) = \iiint f_M(x, y, t) \exp[-j2\pi(ux + vy + wt)] dx dy dt$$

$$\begin{aligned} F_M(u, v, w) &= \iiint f_0(x - s_x t, y - s_y t) \exp[-j2\pi(ux + vy + wt)] dx dy dt \\ &= F_0(u, v) \cdot \delta(us_x + vs_y + w) \end{aligned}$$

时空频谱的定义域是平面

$$us_x + vs_y + w = 0$$



14.4.2 基于运动补偿的滤波

2. 沿运动轨迹的滤波

脉冲响应

$$h(x, y, t) = h_1(t) \delta(x - s_x t, y - s_y t)$$

$$\begin{aligned} H(u, v, w) &= \iiint h_1(t) \delta(x - s_x t, y - s_y t) \exp[-j2\pi(ux + vy + wt)] dx dy dt \\ &= \int h_1(t) \exp[-j2\pi(us_x + vs_y + w)t] dt = H_1(\underbrace{us_x + vs_y + w}) \end{aligned}$$

没有运动补偿时的纯时间滤波情况

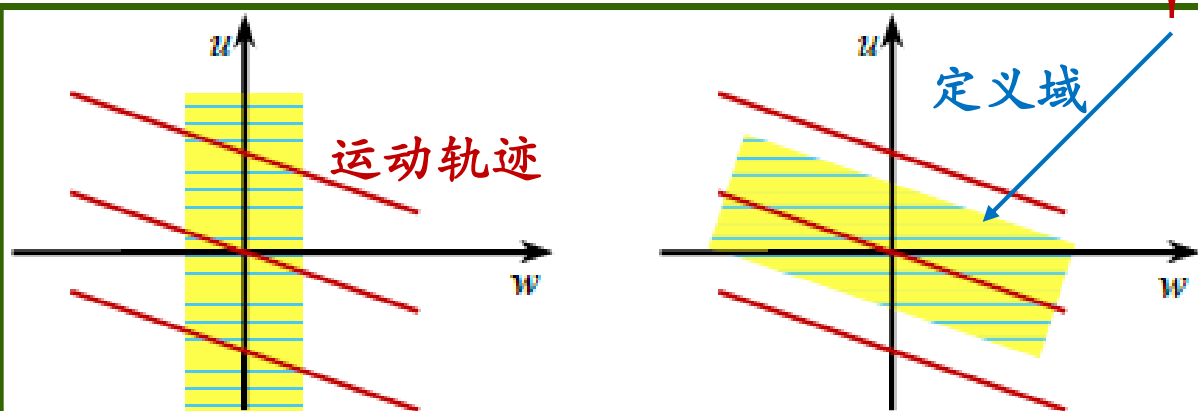


图 14.4.3 运动补偿滤波器的频率响应的定义域

有运动补偿且正确时的情况

14.4.2 基于运动补偿的滤波

3. 运动补偿滤波器

在时空域的采样序列中对运动轨迹估计
使用 N 帧图来对第 k 帧图进行滤波

在第 k 帧的 (x, y) 处估计离散运动轨迹 $M(l; x, y, t)$,
它给出第 k 帧图中 (x, y) 处像素对应的第 l 帧图中的坐标

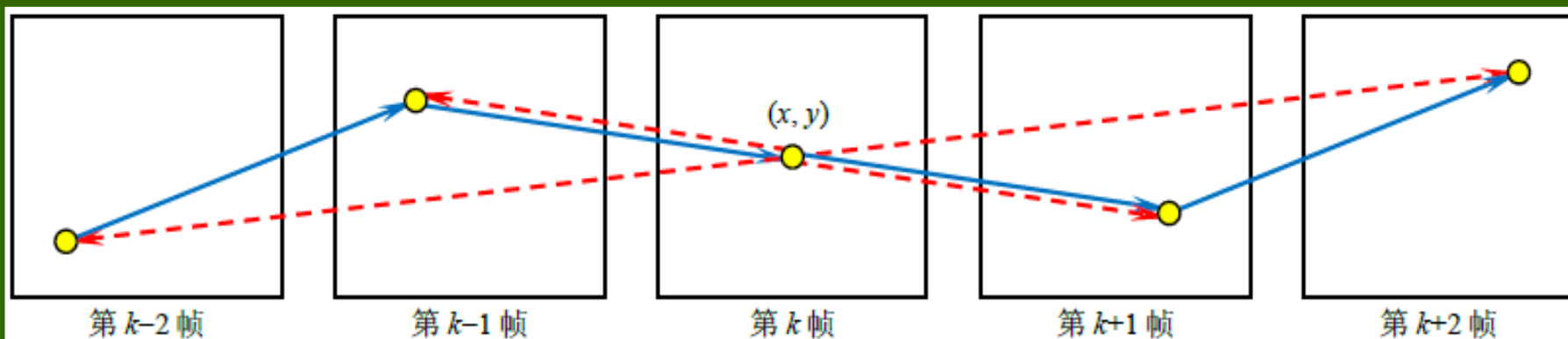


图 14.4.4 运动轨迹估计

14.4.2 基于运动补偿的滤波

4. 时空自适应线性最小均方误差滤波

在 (x, y, t) 处的像素估计值 (考虑平稳独立噪声)

$$\hat{f}(x, y, t) = \frac{\sigma_f^2(x, y, t)}{\sigma_f^2(x, y, t) + \sigma_n^2(x, y, t)} g(x, y, t) + \frac{\sigma_n^2(x, y, t)}{\sigma_f^2(x, y, t) + \sigma_n^2(x, y, t)} \mu_g(x, y, t)$$

滤波器的自适应能力

(1) 时空信号的方差远小于噪声方差时， $\sigma_f^2(x, y, t) \approx 0$ ，估计值逼近时空均值， $\mu_g (= \mu_f)$

(2) 时空信号的方差远大于噪声方差时， $\sigma_f^2(x, y, t) \gg \sigma_n^2(x, y, t)$ ，估计值逼近噪声图象值 g

14.4.3 消除匀速直线运动模糊

匀速直线运动

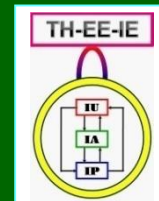
T: 采集时间长度

x 方向运动分量

y 方向运动分量

$$g(x, y) = \int_0^T f[x - x_0(t), y - y_0(t)] dt$$

$$\begin{aligned} G(u, v) &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} g(x, y) \exp[-j2\pi(ux + vy)] dx dy \\ &= \int_0^T \left[\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f[x - x_0(t), y - y_0(t)] \exp[-j2\pi(ux + vy)] dx dy \right] dt \\ &= F(u, v) \int_0^T \exp\{-j2\pi[ux_0(t) + vy_0(t)]\} dt = H(u, v) F(u, v) \end{aligned}$$



14.4.3 消除匀速直线运动模糊

匀速直线运动

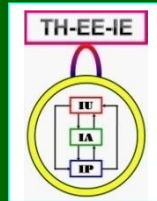
定义滤波函数

$$H(u, v) = \int_0^T \exp \{-j2\pi[ux_0(t) + vy_0(t)]\} dt$$

适合使用逆滤波方法

$$G(u, v) = H(u, v)F(u, v)$$

当景物在平面上的匀速运动沿直线进行时， $x_0(t)$ 和 $y_0(t)$ 均是 t 的线性函数（且可从模糊视频中估计出来），滤波函数中的积分可解析计算，由运动造成的模糊就可以消除



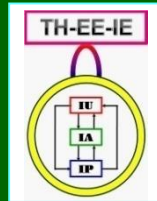
14.5 视频预测编码

在视频中，不仅在**帧内**可使用预测编码方法（如同静止图象），而且在**帧间**也可使用预测编码方法，即利用前后帧之间的相关性

1. 预测基础

采用基于子图象（视频中一般称为**宏块**）的正交变换编码方法进行帧内编码，这样得到的编码帧称为**独立帧**（I-frame）

独立帧在编解码中都不依赖于其他帧，也适合于作为进行预测编码的起始帧



14.5 视频预测编码

2. 单向时间预测

基于独立帧，可对其相邻帧进行预测编码
线性预测（帧间没有运动变化）

$$f(x, y, t) = f(x, y, t - 1)$$

运动补偿预测（MCP）

$$f_p(x, y, t) = f(x + dx, y + dy, t - 1)$$

参考帧 $f(x, y, t-1)$ 须在编码帧 $f(x, y, t)$ 前被编码并重建，而编码帧要借助预测帧 $f_p(x, y, t)$ 进行编码

14.5 视频预测编码

2. 单向时间预测

- (1) I帧：每组的第一帧， \Rightarrow 帧内编码
- (2) P帧：同组的剩余帧，对它们进行预测编码， \Rightarrow 帧间编码

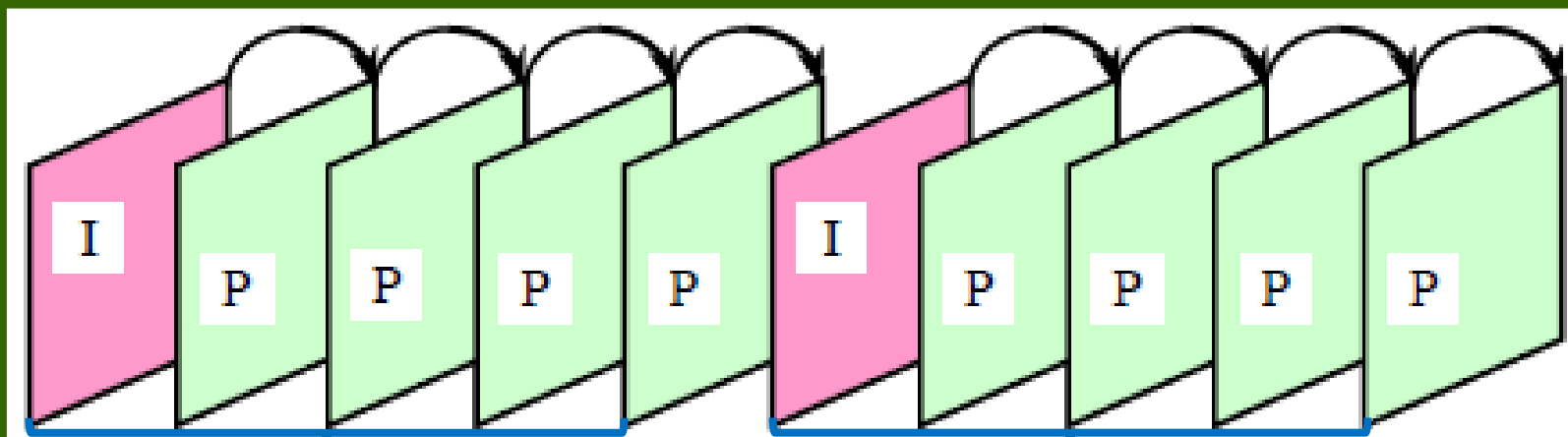


图 14.5.1 单向时间预测序列示意图 (H.261)

14.5 视频预测编码

3. 双向时间预测

当前帧内的一个像素值可结合其前一帧的像素和后一帧的像素来预测

$$f_p(x, y, t) = af(x + d_ax, y + d_ay, t - 1) + bf(x - d_bx, y - d_by, t + 1)$$

用 $f(x, y, t-1)$ 来预测 $f(x, y, t)$, 称为前向预测

(d_ax, d_ay) 表示从 $t-1$ 到 t 的运动矢量

用 $f(x, y, t+1)$ 来预测 $f(x, y, t)$, 称为后向预测

(d_bx, d_by) 表示从 t 到 $t+1$ 的运动矢量

14.5 视频预测编码

3. 双向时间预测

- (1) I帧：仅进行帧内编码
- (2) P帧：参照前一幅I帧或P帧进行帧间编码
- (3) B帧：也称双向预测帧，压缩率最大

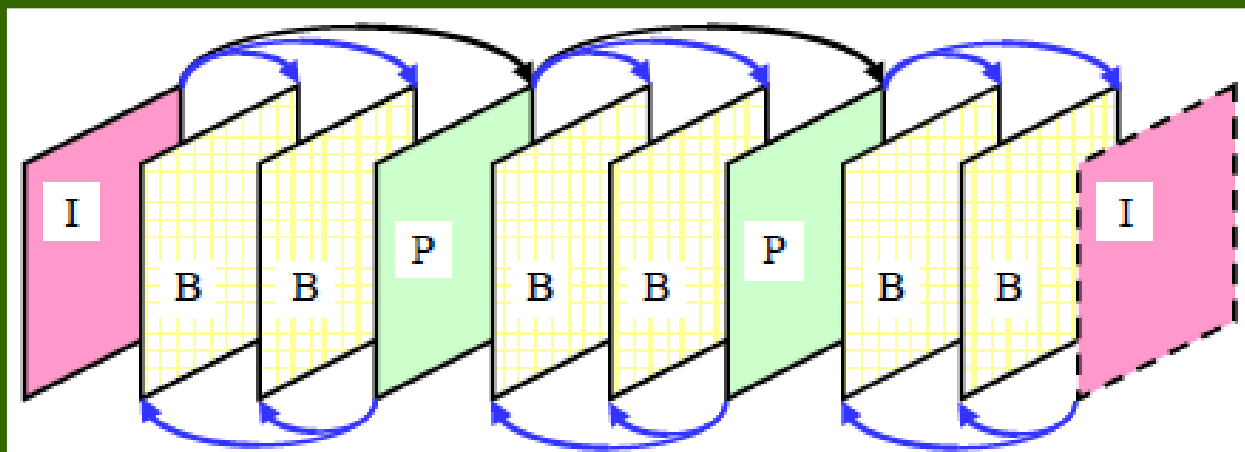


图 14.5.2 双向时间预测序列示意图 (MPEG-1)



联系信息

- ☞ 通信地址：北京清华大学电子工程系
- ☞ 邮政编码：100084
- ☞ 办公地址：清华大学，罗姆楼，6层305室
- ☞ 办公电话：(010) 62798540
- ☞ 传真号码：(010) 62770317
- ☞ 电子邮件：zhang-yj@tsinghua.edu.cn
- ☞ 个人主页：oa.ee.tsinghua.edu.cn/~zhangyujin/