

文章编号:1671-1637(2012)04-0100-07

## 运动车辆检测的 APG-TR 算法

陈涛<sup>1</sup>, 谭华春<sup>2</sup>, 冯广东<sup>2</sup>, 王震宇<sup>3</sup>, 魏朗<sup>1</sup>

(1. 长安大学 汽车运输安全保障技术交通行业重点实验室, 陕西 西安 710064; 2. 北京理工大学 机械与车辆学院, 北京 100081; 3. 南佛罗里达大学 城市交通研究中心, 佛罗里达 坦帕 33620)

**摘要:** 为了提高智能交通系统中运动车辆检测的准确率, 提出了一种基于张量恢复的 APG-TR 算法。采用张量表征交通视频图像, 保持视频图像高维结构特征。通过张量恢复, 重建出张量的低秩部分与稀疏部分, 实现交通视频图像中交通背景与运动目标车辆的分离与交通视频内在特征的提取。利用交通监控系统采集到的交通视频 106 帧图像对本文算法进行了测试。测试结果表明: 在晴天条件下, APG-TR 算法的平均正确率为 91.4%, 在雨、雾天气条件下, 正确率分别为 86.4%、85.2%, 相比帧差法更加稳定与准确。APG-TR 算法具有良好的收敛速度与鲁棒性, 在智能交通领域中具有广泛的应用前景。

**关键词:** 智能交通系统; 车辆检测; 高维结构; 张量恢复; APG-TR; 矩阵恢复

**中图分类号:** U491.116

**文献标志码:** A

## APG-TR algorithm of moving vehicle detection

CHEN Tao<sup>1</sup>, TAN Hua-chun<sup>2</sup>, FENG Guang-dong<sup>2</sup>, WANG Zhen-yu<sup>3</sup>, WEI Lang<sup>1</sup>

(1. Key Laboratory of Automotive Transportation Safety Technology of Ministry of Transport, Chang'an University, Xi'an 710064, Shaanxi, China; 2. School of Mechanical and Vehicular Engineering, Beijing Institute of Technology, Beijing 100081, China; 3. Center for Urban Transportation Research, University of South Florida, Tampa 33620, Florida, USA)

**Abstract:** In order to improve the accuracy of moving vehicle detection in intelligent transportation system, an accelerated proximal gradient-tensor recovery (APG-TR) algorithm was proposed based on tensor recovery. The traffic video image data were characterized by using tensor in the algorithm, which maintained the high-dimensional structure characteristic of video image. The lower rank part and sparse part in the tensor were effectively reconstructed by tensor recovery, and moving target vehicle and traffic background were separated, therefore the internal properties were easily extracted. The algorithm was tested by using 106 video images collected by traffic monitoring system. Test result shows that the average detection accuracies are 91.4% in fine days, 86.4% and 85.2% under rain and fog conditions respectively, which are more stable and accurate compared with the frame differential method. APG-TR algorithm is proved to have good convergence speed and robust, and has abroad application in the field of intelligent transportation. 3 figs, 18 refs.

**Key words:** ITS; vehicle detection; high-dimensional structure; tensor recovery; APG-TR; matrix recovery

收稿日期: 2012-02-09

基金项目: 国家自然科学基金项目 (50908020); 北京市自然科学基金项目 (4122067); 中央高校基本科研业务费专项资金项目 (CHD2012ZD018, CHD2011SY010)

作者简介: 陈涛 (1974-), 男, 陕西铜川人, 长安大学副教授, 工学博士, 从事交通安全与人车路系统动力学研究。

**Author resume:** CHEN Tao(1974-), male, associate professor, PhD, +86-29-82334475, chentao@chd.edu.cn.

## 0 引言

智能交通运输系统(ITS)已经成为交通领域研究的热点,深受社会各界的关注,通过智能交通运输系统来缓解日趋严重的交通问题迫在眉睫<sup>[1]</sup>。在ITS的众多交通信息中,车辆运行信息最具有代表性,为交通管理部门制定政策并采取措施提供了科学客观的依据<sup>[2]</sup>,因此,在进行交通系统的智能控制方面,如何快速、准确获取车辆的运行信息是急需解决的重要问题之一。

目前,国内外常用的交通场景中运动车辆的检测方法主要有背景帧差法<sup>[3]</sup>、模板匹配法<sup>[4]</sup>、支持向量机法<sup>[5]</sup>、光流法<sup>[6]</sup>等。背景帧差法具有实时性好,模板更新速度快等特点,但是在实际的交通场景中,交通背景往往十分复杂,同时运动车辆之间经常会相互遮挡,因此,该方法很难从复杂的交通背景中准确地提取运动车辆信息。模板匹配法仅适应于较短时间内的车辆检测,随着运动车辆位置的不断变化,累计误差将会逐渐增大,导致车辆检测失效,鲁棒性变差。支持向量机(Support Vector Machine, SVM)法通过有限的样本信息,选择出该条件下的最佳方案,适应于小样本、非线性的车辆目标识别。然而,由于实际道路中的车辆种类繁多,对于大规模的训练样本,SVM法难以实施,同时,随着样本数量的增加,训练速度变慢,导致实时性变差。光流法不需要提供交通场景的先验知识就能够完成独立运动车辆的检测,但是其算法复杂,运算量较大,很难满足实际运用中对实时性的要求。

综上所述,上述常用算法还存在局限性,在计算效率与可靠性方面还存在问题。针对复杂道路场景的运动车辆检测,本文将其转化为基于张量模型的低秩恢复问题,首先将多帧视频构建为三维张量模型,由于视频帧之间的背景信息变化不大,可近似为低秩的,而前方运动车辆信息可近似为稀疏的,通过对低秩与稀疏特性的约束,可以很好地分离出背景及运动车辆信息,进而实现车辆的检测。相对于常用方法,本文方法从数学模型角度解决车辆的检测问题,可以很好地去除阴影、光照以及车辆遮挡所带来的干扰,具有很好的鲁棒性。同时优化算法中使用了加速近端梯度乘数(Accelerated Proximal Gradient, APG)算法,大大

提高了运算速度<sup>[7]</sup>,因此,对运动车辆的检测具有很大的应用价值与现实意义。

## 1 APG-TR 算法

张量为多维数组,是向量与矩阵的高阶推广形式,广泛应用于信息科学、计算机视觉及图形分析等领域。Wright等证明了如果不规则的稀疏矩阵 $S_m$ 保持稀疏特性(相对于低秩矩阵 $L_m$ )<sup>[8]</sup>,可以通过解决式(1)所述的凸优化问题,实现稀疏及低秩部分的分离,该优化方法被定义为鲁棒主成分分析法(RP-CA)或主成分追踪法(PCP)<sup>[9]</sup>

$$\begin{cases} \min G(L_m, S_m) = \|L_m\|_* + \eta \|S_m\|_1 \\ A_m = L_m + S_m \end{cases} \quad (1)$$

式中: $G(\cdot)$ 为凸优化目标函数; $A_m$ 为给定的待恢复矩阵; $L_m$ 为矩阵 $A_m$ 的低秩部分; $S_m$ 为矩阵 $A_m$ 的稀疏部分; $\eta$ 为大于0的系数。

在交通视频中可通过交通背景建模来提取运动车辆,通常采用二维数据矩阵来表征视频中的每幅图像。实际上,交通视频为三维数据,采用二维数据结构损坏了图像的原始结构,在分析过程中无法充分挖掘其内部特性。鉴于张量形式能保持交通图像的原始结构与内部特性,本文采用张量模型来表示交通图像,进而构建基于张量模型的运动车辆检测方法。本文方法结合张量恢复理论,将式(1)扩展到张量形式,构建张量恢复目标函数,用低秩部分表示交通背景,稀疏部分表示运动车辆,通过背景建模方法实现运动车辆的检测。

Li等首先提出了张量恢复方法(RSTD),采用张量迹范数近似张量的秩与1范数近似 $l_1$ 范数,解决了“非凸”的困难<sup>[10]</sup>,但是该方法收敛慢,精度不高。在矩阵恢复中,APG算法能很好地松弛目标函数,消除相互依赖的迹范数与 $l_1$ 范数约束,因此,本文将APG算法推广到张量恢复中,构建了基于APG的张量恢复方法。将矩阵主成分跟踪问题通过凸优化向张量扩展,并且充分利用良好的分割结构实现张量恢复问题的算法设计。本文算法主要针对 $n$ 阶张量的恢复,称为APG-TR(Accelerated Proximal Gradient-Tensor Recovery)算法,与RSTD相比,采取了松弛策略将张量恢复问题转化为混合凸优化问题,消除了相互依赖的迹范数与 $l_1$ 范数约束<sup>[11]</sup>,因此,可以将APG-TR算法看作是一

个多线性扩展的 RPCA 问题,包含了矩阵 RPCA 问题这一个特殊例子。Zhou 等证明该矩阵 RPCA 问题可以通过 APG 算法解决,以达到较高的精度,较小的内存要求与令人满意的收敛速度<sup>[12]</sup>。本文将 APG 问题扩展为多线性 RPCA 问题,并且证明了 APG 算法不仅适合解决矩阵 RPCA 问题,也适合解决多线性 RPCA 问题。

### 1.1 张量恢复

Candes 等提出了矩阵恢复问题<sup>[13]</sup>,即最小化目标函数  $G(\mathbf{L}_m, \mathbf{S}_m)$

$$\begin{cases} \min G(\mathbf{L}_m, \mathbf{S}_m) = \|\mathbf{L}_m\|_* + \eta \|\mathbf{S}_m\|_1 \\ \text{s. t. } \|\mathbf{A}_m - \mathbf{L}_m - \mathbf{S}_m\|_F \leq \varepsilon \end{cases} \quad (2)$$

将式(2)中的矩阵扩展到张量形式,即

$$\begin{cases} \min G(\mathbf{L}, \mathbf{S}) = \|\mathbf{L}\|_* + \eta \|\mathbf{S}\|_1 \\ \text{s. t. } \|\mathbf{A} - \mathbf{L} - \mathbf{S}\|_F \leq \varepsilon \end{cases} \quad (3)$$

式中: $\mathbf{A}, \mathbf{L}, \mathbf{S}$  为与  $\mathbf{A}_m, \mathbf{L}_m, \mathbf{S}_m$  具有相同形状与大小的  $n$  阶张量; $\varepsilon$  为任意充分小的正数。

式(3)可以转化为凸优化形式<sup>[14]</sup>

$$\begin{cases} \min G(\mathbf{L}, \mathbf{S}) = \sum_{k=1}^K \gamma_k \|\mathbf{L}_k\|_* + \sum_{k=1}^K \eta_k \|\mathbf{S}_k\|_1 \\ \text{s. t. } \|\mathbf{A} - \mathbf{L} - \mathbf{S}\|_F^2 < \varepsilon \end{cases} \quad (4)$$

式中: $\mathbf{L}_k, \mathbf{S}_k$  分别为按第  $k$  模式展开的张量; $K$  为张量  $\mathbf{L}, \mathbf{S}$  的阶数; $\gamma_k, \eta_k$  分别为按第  $k$  模式展开后范数的权重参数。

为了简化问题,引入辅助矩阵  $\mathbf{P}_k, \mathbf{Z}_k$

$$\begin{cases} \mathbf{P}_k = \mathbf{L}_k \\ \mathbf{Z}_k = \mathbf{S}_k \end{cases} \quad (5)$$

得到

$$\begin{cases} \min G(\mathbf{L}, \mathbf{S}, \mathbf{P}_k, \mathbf{Z}_k) = \sum_{k=1}^K \gamma_k \|\mathbf{P}_k\|_* + \sum_{k=1}^K \eta_k \|\mathbf{Z}_k\|_1 \\ \text{s. t. } \|\mathbf{A}_k - \mathbf{P}_k - \mathbf{Z}_k\|_F^2 < \varepsilon \end{cases} \quad (6)$$

式中: $\mathbf{A}_k$  为按第  $k$  模式展开的张量。

进一步通过  $\|\mathbf{P}_k - \mathbf{L}_k\|_F \leq \varepsilon$  与  $\|\mathbf{Z}_k - \mathbf{S}_k\|_F \leq \varepsilon$  松弛等价约束式(5),得到松弛后的目标函数为

$$\begin{cases} \min G(\mathbf{L}, \mathbf{S}, \mathbf{P}_k, \mathbf{Z}_k) = \sum_{k=1}^K \gamma_k \|\mathbf{P}_k\|_* + \sum_{k=1}^K \eta_k \|\mathbf{Z}_k\|_1 \\ \text{s. t. } \|\mathbf{P}_k - \mathbf{L}_k\|_F^2 < \varepsilon \\ \|\mathbf{Z}_k - \mathbf{S}_k\|_F^2 < \varepsilon \\ \|\mathbf{A}_k - \mathbf{P}_k - \mathbf{Z}_k\|_F^2 < \varepsilon \end{cases} \quad (7)$$

### 1.2 经典 APG 方法

APG 算法为利用 Nesterov 技巧的一阶算法,其收敛速度很有竞争力,可以解决如下优化问题

$$\begin{cases} \min_{\mathbf{x} \in \mathbf{H}} z = g(\mathbf{x}) \\ \text{s. t. } B(\mathbf{x}) = \mathbf{b} \end{cases} \quad (8)$$

式中: $\mathbf{H}$  为空间范数; $z$  为优化目标函数; $g(\mathbf{x})$  为一个连续的凸函数; $B(\cdot)$  为一个线性映射; $\mathbf{b}$  为一个观察量; $\mathbf{x}$  为自变量。

对式(8)进行拉格朗日变换可得<sup>[15-16]</sup>

$$\begin{cases} \min_{\mathbf{x} \in \mathbf{H}} F(\mathbf{x}) = \mu g(\mathbf{x}) + f(\mathbf{x}) \\ f(\mathbf{x}) = \frac{1}{2} \|B(\mathbf{x}) - \mathbf{b}\|^2 \end{cases} \quad (9)$$

式中: $F(\mathbf{x})$  为目标函数; $f(\mathbf{x})$  为辅助函数; $\mu$  为拉格朗日系数。

APG 算法每次最小化  $F(\mathbf{x})$  在某个点  $\mathbf{y}$  处的一个二阶近似函数,即

$$\begin{aligned} Q(\mathbf{x}, \mathbf{y}) &= f(\mathbf{y}) + \langle \nabla f(\mathbf{x}), \mathbf{x} - \mathbf{y} \rangle + \\ &\quad \frac{L_f}{2} \|\mathbf{x} - \mathbf{y}\|^2 + \mu g(\mathbf{x}) \end{aligned} \quad (10)$$

式中: $Q(\mathbf{x}, \mathbf{y})$  为  $F(\mathbf{x})$  在  $\mathbf{y}$  处的二阶近似函数; $\mathbf{y}$  为自变量; $\nabla$  为向后差分算子; $\langle \cdot \rangle$  为内积操作; $L_f$  为  $\nabla f(\mathbf{x})$  的 Lipschitz 常数。

定义辅助变量  $\mathbf{G}_1$  为

$$\mathbf{G}_1 = \mathbf{y} - \frac{1}{L_f} \nabla f(\mathbf{y}) \quad (11)$$

则有

$$\begin{aligned} \arg \min [Q(\mathbf{x}, \mathbf{y})] &= \arg \min [\mu g(\mathbf{x}) + \\ &\quad \frac{L_f}{2} \|\mathbf{x} - \mathbf{G}_1\|^2] \end{aligned} \quad (12)$$

为求解式(12),反复设置

$$x_{h+1} = \arg \min [Q(x_h, y_h)]$$

式中: $x_h, y_h$  为  $\mathbf{x}, \mathbf{y}$  的迭代序列值, $h$  为迭代序号。

在光滑情况下  $g(\mathbf{x})=0$ ,设置

$$y_h = x_h + \frac{t_{h-1} - 1}{t_h} (x_h - x_{h-1})$$

当满足下式时,能提高收敛速度<sup>[14]</sup>

$$t_{h+1}^2 - t_{h+1} \leq t_h^2$$

式中: $t_h$  为  $t$  的迭代序列值。

APG 算法具体步骤如下。

$$(1) y_h \leftarrow x_h + \frac{t_{h-1} - 1}{t_h} (x_h - x_{h-1})$$

$$(2) \mathbf{G}_1 \leftarrow \mathbf{y} - \frac{1}{L_f} \nabla f(\mathbf{y})$$

$$(3) x_{h+1} \leftarrow \arg \min [\mu g(\mathbf{x}) + \frac{L_f}{2} \|\mathbf{x} - \mathbf{G}_1\|^2]$$

$$(4) t_{h+1} \leftarrow \frac{1 + \sqrt{4t_h^2}}{2}, h \leftarrow h + 1$$

(5)若迭代次数大于阈值或  $t_{h+1} - t_h$  小于收敛阈值  $\Delta$ ,则循环结束,否则转步骤(2)。

### 1.3 将APG扩展到张量恢复

将求解矩阵问题的APG算法推广到张量上,参考式(7),给定张量 $\mathbf{X}, \mathbf{Y}$ ,可以得到

$$Q(\mathbf{X}, \mathbf{Y}) = Q(\mathbf{X}^M, \mathbf{X}^N, \mathbf{Y}^M, \mathbf{Y}^N) = \sum_{k=1}^K \left[ \left( \frac{\gamma_k}{2} \|\mathbf{X}_k^M - \mathbf{Y}_k^M + \frac{1}{L_f} (\mathbf{Y}_k^M + \mathbf{Y}_k^N - \mathbf{A}_k)\|_F^2 + \frac{\alpha_k}{2} \|\mathbf{X}_k^L - \mathbf{X}_k^M\|_F^2 + \lambda_k \|\mathbf{X}_k^M\|_* \right) + \left( \frac{\gamma_k}{2} \|\mathbf{X}_k^N - \mathbf{Y}_k^N + \frac{1}{L_f} (\mathbf{Y}_k^M + \mathbf{Y}_k^N - \mathbf{A}_k)\|_F^2 + \frac{\beta_k}{2} \|\mathbf{X}_k^S - \mathbf{X}_k^N\|_1 + \eta_k \|\mathbf{X}_k^N\|_1 \right) \right] \quad (13)$$

式中: $Q(\mathbf{X}^M, \mathbf{X}^N, \mathbf{Y}^M, \mathbf{Y}^N)$ 为 $Q(\mathbf{X}, \mathbf{Y})$ 的等价函数; $\alpha_k, \beta_k$ 为权重系数; $\mathbf{X}^M, \mathbf{Y}^M$ 分别为张量 $\mathbf{X}, \mathbf{Y}$ 的低秩部分; $\mathbf{X}^N, \mathbf{Y}^N$ 分别为张量 $\mathbf{X}, \mathbf{Y}$ 的稀疏部分; $\mathbf{X}_k^M, \mathbf{X}_k^N, \mathbf{Y}_k^M, \mathbf{Y}_k^N$ 分别为张量 $\mathbf{X}^M, \mathbf{X}^N, \mathbf{Y}^M, \mathbf{Y}^N$ 在第 $k$ 模式的展开; $\mathbf{X}_k^L$ 为张量 $\mathbf{A}$ 的低秩部分 $\mathbf{L}_k$ 按第 $k$ 模式的展开; $\mathbf{X}_k^S$ 为张量 $\mathbf{A}$ 的稀疏部分 $\mathbf{S}_k$ 按第 $k$ 模式的展开; $\lambda_k$ 为化简后的常数。

采用块协调下降(BCD)算法求解式(13)<sup>[17]</sup>,具体步骤如下。

#### (1) 计算 $\mathbf{X}_N^*$

固定其余变量,最小化自变量为 $\mathbf{X}_k^N$ 的目标函数 $G(\mathbf{X}_k^N)$ ,即

$$\min G(\mathbf{X}_k^N) = \frac{\gamma_k}{2} \left\| \mathbf{X}_k^N - \mathbf{Y}_k^N + \frac{1}{L_f} (\mathbf{Y}_k^M + \mathbf{Y}_k^N - \mathbf{A}_k) \right\|_F^2 + \frac{\beta_k}{2} \left\| \mathbf{X}_k^S - \mathbf{X}_k^N \right\|_1 + \eta_k \|\mathbf{X}_k^N\|_1 \quad (14)$$

式(14)最优问题的全局最小值 $\mathbf{X}_N^*$ 为<sup>[18]</sup>

$$\begin{cases} \mathbf{X}_N^* = D \left\{ \frac{\beta_k \mathbf{X}_k^S + \gamma_k [\mathbf{Y}_k^N - (\mathbf{Y}_k^M + \mathbf{Y}_k^N - \mathbf{A}_k)/L_f]}{\gamma_k + \beta_k}, \right. \\ \left. \frac{\eta_k}{\gamma_k + \beta_k} \right\} \\ e = \frac{\beta_k \mathbf{X}_k^S + \gamma_k [\mathbf{Y}_k^N - (\mathbf{Y}_k^M + \mathbf{Y}_k^N - \mathbf{A}_k)/L_f]}{\gamma_k + \beta_k} \\ \tau = \frac{\eta_k}{\gamma_k + \beta_k} \\ D(e, \tau) = \begin{cases} e - \tau & e > \tau \\ e + \tau & e < \tau \\ 0 & e = \tau \end{cases} \end{cases} \quad (15)$$

式中: $D(\cdot)$ 为收缩操作函数。

#### (2) 计算 $\mathbf{X}_M^*$

固定其余变量,最小化自变量为 $\mathbf{X}_k^M$ 的目标函数 $G(\mathbf{X}_k^M)$ ,即

$$\min G(\mathbf{X}_k^M) = \frac{\gamma_k}{2} \left\| \mathbf{X}_k^M - \mathbf{Y}_k^M + \frac{1}{L_f} (\mathbf{Y}_k^M + \mathbf{Y}_k^N - \mathbf{A}_k) \right\|_F^2 + \frac{\alpha_k}{2} \left\| \mathbf{X}_k^L - \mathbf{X}_k^M \right\|_F^2 + \lambda_k \|\mathbf{X}_k^M\|_* \quad (16)$$

式(16)的全局最小值 $\mathbf{X}_M^*$ 为<sup>[19]</sup>

$$\mathbf{X}_M^* = \mathbf{U}_k D \left( \mathbf{A}, \frac{\lambda_k}{\gamma_k + \alpha_k} \right) \mathbf{V}_k^T \quad (17)$$

$$R(\mathbf{U}_k, \mathbf{V}_k^T) = \frac{\alpha_k \mathbf{X}_k^L + \gamma_k [\mathbf{Y}_k^M - (\mathbf{Y}_k^M + \mathbf{Y}_k^N - \mathbf{A}_k)/L_f]}{\gamma_k + \alpha_k} \quad (18)$$

式中: $\mathbf{A}$ 为对角矩阵; $R(\mathbf{U}_k, \mathbf{V}_k^T)$ 为执行矩阵奇异值分解, $\mathbf{U}_k, \mathbf{V}_k^T$ 为分解后得到的矩阵。

#### (3) 计算 $\mathbf{S}^*$

固定其余变量,最小化自变量为 $\mathbf{S}$ 的目标函数 $G(\mathbf{S})$ ,即

$$\min G(\mathbf{S}) = \sum_{k=1}^K \frac{\beta_k}{2} \left\| \mathbf{X}_k^S - \mathbf{X}_k^N \right\|_F^2 \quad (19)$$

求解式(19),得到最优解 $\mathbf{S}^*$ 为

$$\mathbf{S}^* = \frac{\sum_{k=1}^K \beta_k \mathbf{p}(\mathbf{X}_k^N, k)}{\sum_{k=1}^K \beta_k} \quad (20)$$

式中: $\mathbf{p}(\mathbf{X}_k^N, k)$ 为将矩阵 $\mathbf{X}_k^N$ 按第 $k$ 模式变换的张量。

#### (4) 计算 $\mathbf{L}^*$

固定其余变量,最小化自变量为 $\mathbf{L}$ 的目标函数 $G(\mathbf{L})$ ,得

$$\min G(\mathbf{L}) = \sum_{k=1}^K \frac{\alpha_k}{2} \left\| \mathbf{X}_k^L - \mathbf{X}_k^M \right\|_F^2 \quad (21)$$

求解式(21),得到最优解 $\mathbf{L}^*$ 为

$$\mathbf{L}^* = \frac{\sum_{k=1}^K \alpha_k \mathbf{p}(\mathbf{X}_k^M, k)}{\sum_{k=1}^K \alpha_k} \quad (22)$$

连续迭代过程中,选择 $\mathbf{L}$ 与 $\mathbf{S}$ 之间的差在某个误差容限内作为停止标准。反复估算 $\mathbf{X}_k^N, \mathbf{X}_k^M, \mathbf{L}^*, \mathbf{S}^*$ ,直到收敛。

在APG-TR的伪代码算法中:输入为 $n$ 阶张量 $\mathbf{A}$ ;参数为 $\alpha_k, \beta_k, \gamma_k, \eta_k, \lambda_k, m$ ,其中 $m$ 为外循环控制次数;输出为 $n$ 阶张量 $\mathbf{L}, \mathbf{S}$ 。算法步骤如下。

(1) 令 $\mathbf{L}^{(0)}$ 为 $\mathbf{A}, \mathbf{S}^{(0)}$ 为 $\mathbf{0}, k$ 为1。

(2) 外部循环开始, $m=1$ 。

(3) 内部循环开始, $k$ 从1循环至 $n$ 。

(4) 求解 $\mathbf{X}_N^*$

$$\mathbf{Y}_k^N \leftarrow \mathbf{X}_k^N + \frac{t_{k-1} - 1}{t_k} (\mathbf{X}_k^N - \mathbf{X}_{k-1}^N)$$

$$\mathbf{X}_N^* = D \left\{ \frac{\beta_k \mathbf{X}^S + \gamma_k [\mathbf{Y}_k^N - (\mathbf{Y}_k^M + \mathbf{Y}_k^N - \mathbf{A}_k) / L_f]}{\gamma_k + \beta_k}, \frac{\eta_k}{\gamma_k + \beta_k} \right\}$$

(5) 求解  $\mathbf{X}_M^*$

$$\mathbf{Y}_k^M \leftarrow \mathbf{X}_k^M + \frac{t_{k-1} - 1}{t_k} (\mathbf{X}_k^M - \mathbf{X}_{k-1}^M)$$

$$\mathbf{X}_M^* = U_k D \left( \mathbf{A}, \frac{\lambda_k}{\gamma_k + \alpha_k} \right) \mathbf{V}_k^T$$

$$U_k \mathbf{A} \mathbf{V}_k^T = \frac{\alpha_k \mathbf{X}^L + \gamma_k [\mathbf{Y}_k^M - (\mathbf{Y}_k^M + \mathbf{Y}_k^N - \mathbf{A}_k) / L_f]}{\gamma_k + \alpha_k}$$

(6) 内部循环结束。

(7) 求解第  $m$  次外循环后得到的  $\mathbf{S}^*$ 、 $\mathbf{L}^*$  的值

$\mathbf{S}^{(m)}$ 、 $\mathbf{L}^{(m)}$

$$\left\{ \begin{array}{l} \mathbf{S}^{(m)} = \frac{\sum_{k=1}^K \beta_k \mathbf{p}(\mathbf{X}_k^N, k)}{\sum_{k=1}^K \beta_k} \\ \mathbf{L}^{(m)} = \frac{\sum_{k=1}^K \alpha_k \mathbf{p}(\mathbf{X}_k^M, k)}{\sum_{k=1}^K \alpha_k} \end{array} \right.$$

(8) 如果迭代次数大于阈值,或同时满足下式,则外部循环结束,否则  $m = m + 1$ ,转至步骤(3)

$$\left| \frac{\mathbf{S}^{(m+1)} - \mathbf{S}^{(m)}}{\mathbf{S}^{(m+1)}} \right| < \Delta$$

$$\left| \frac{\mathbf{L}^{(m+1)} - \mathbf{L}^{(m)}}{\mathbf{L}^{(m+1)}} \right| < \Delta$$

## 2 运动车辆识别试验

为评估该算法在运动车辆检测方面的有效性与

可靠性,在晴天、雨天与雾天 3 种天气条件下采用实际的交通视频监控系统进行了测试。在晴天条件下,设置视频图像大小为  $320 \times 240$  像素,采用 106 帧图像,构成  $320 \times 240 \times 106$  的三阶张量  $\mathbf{T}$ 。采用本文算法对张量  $\mathbf{T}$  进行分解,得到低阶张量  $\mathbf{L}$  与稀疏张量  $\mathbf{S}$ ,分别对应于交通视频中的背景与运动车辆区域。试验中分别对红、绿、蓝 3 个颜色通道进行处理,再将得到的结果合成为彩色图像。在 APG-TR 算法中,设

$$(\alpha_1, \alpha_2, \alpha_3) = \left( \frac{T_1}{T_{\max}}, \frac{T_2}{T_{\max}}, \frac{T_3}{T_{\max}} \right)$$

$$(\beta_1, \beta_2, \beta_3) = \left( \frac{T_1}{T_{\max}}, \frac{T_2}{T_{\max}}, \frac{T_3}{T_{\max}} \right)$$

$$(\gamma_1, \gamma_2, \gamma_3) = \left( \frac{T_1}{T_{\max}}, \frac{T_2}{T_{\max}}, \frac{T_3}{T_{\max}} \right)$$

$$(\lambda_1, \lambda_2, \lambda_3) = \left( \frac{w_{\max}}{w_1}, \frac{w_{\max}}{w_2}, \frac{w_{\max}}{w_3} \right)$$

$$(\eta_1, \eta_2, \eta_3) = \left( \frac{1}{\sqrt{T_{\max}}}, \frac{1}{\sqrt{T_{\max}}}, \frac{1}{\sqrt{T_{\max}}} \right)$$

$$T_{\max} = \max\{T_1, T_2, T_3\}$$

$$w_{\max} = \max\{w_1, w_2, w_3\}$$

式中:  $T_1$ 、 $T_2$ 、 $T_3$  分别为张量第 1、2、3 模式维度;  $w_1$ 、 $w_2$ 、 $w_3$  分别为  $T_1$ 、 $T_2$ 、 $T_3$  的 95% 奇异值。

所有试验运行与记录在同一台计算机上进行。为了验证本文算法的性能,将其与常用的帧差法进行了比较,图 1~3 分别为第 21、52、95 帧图像的对比结果。可以看出:随着时间的推移,交通的背景逐渐改变,在变化的背景中,本文算法也能很好地提取出背景与前景,较好地适应光照的变化与消除背景



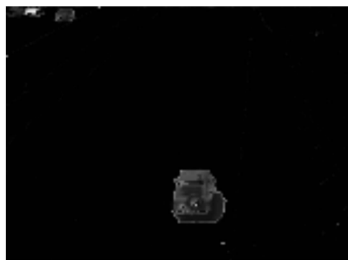
(a) 原始图像



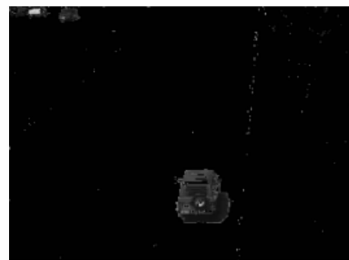
(b) APG-TR恢复背景



(c) 帧差法恢复背景



(d) APG-TR恢复前景



(e) 帧差法恢复前景

图 1 第 21 帧图像的对比结果

Fig. 1 Comparison results of image 21

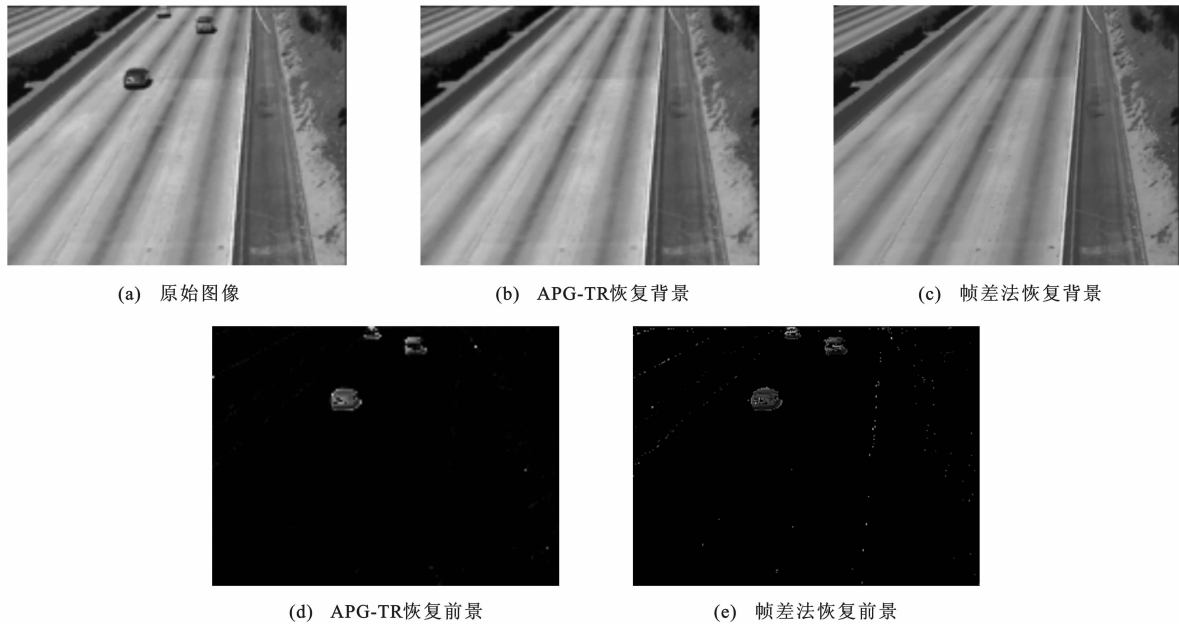


图2 第52帧图像的对比结果

Fig. 2 Comparison results of image 52

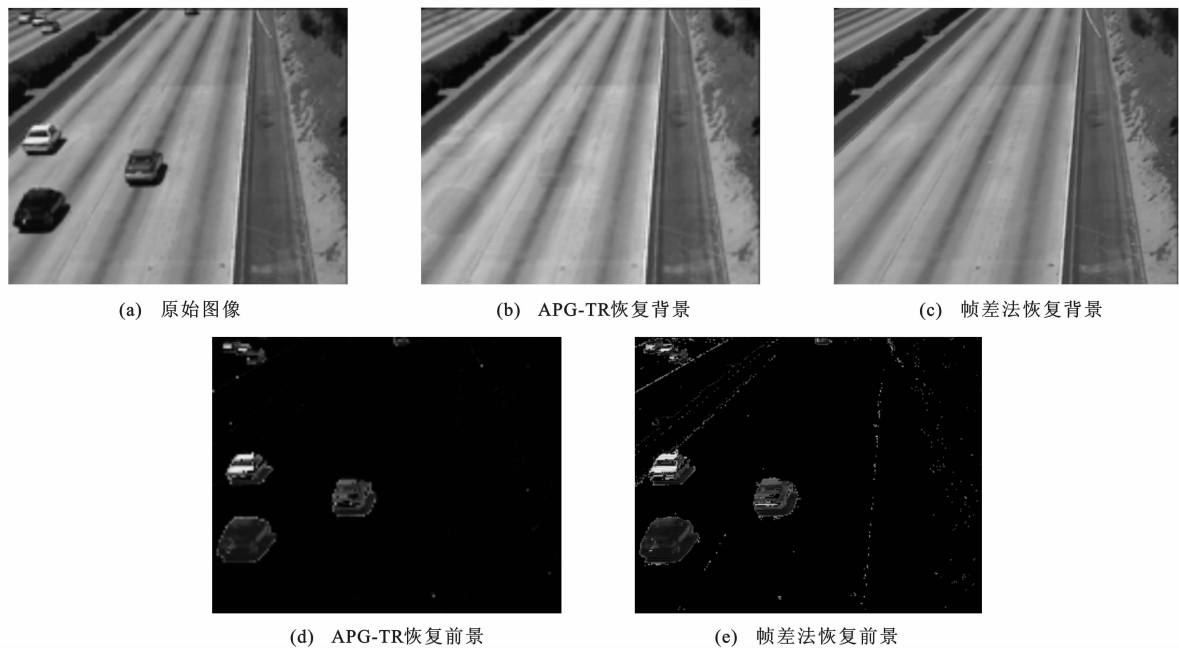


图3 第95帧图像的对比结果

Fig. 3 Comparison results of image 95

光照变化带来的影响(前景中出现的噪声很少);而帧差法则对背景光照的变化非常敏感,导致在提取前景过程中将道路边缘的变化也误认为前景(在道路边缘附近出现噪声)。

为了量化评估算法的检测效果,定义正确率  $c$  为

$$c = 100 \left( \frac{\sum_{s=1}^a s}{\sum_{l=1}^b l} \right)$$

式中:  $a$  为正确检出的运动车辆像素数;  $b$  为实际存在的运动车辆像素数;  $s, l$  为像素计数变量。

分别选取各帧检测结果,计算其平均正确率,进行比较,运动车辆的真实分割结果通过人工获取。在晴天条件下,APG-TR算法的平均正确率为91.4%,帧差法为84.2%。APG-TR算法的检测速度与帧差法相当。为检测不同气候条件下APG-TR算法的鲁棒性,分别对雨天与雾天条件下的交通监控视频

进行了测试,试验设置与前述晴天条件下的试验设置完全相同。利用 APG-TR 算法进行检测,运动车辆的真实分割结果通过人工获取。在雨天与雾天条件下 APG-TR 算法的正确率分别为 86.4%、85.2%,明显优于帧差法 75.2%、73.7%的正确率。可见,APG-TR 算法的运动车辆识别能力明显优于帧差法,并且对天气的适应性更好。

### 3 结 语

(1)提出的 APG-TR 算法可以快速有效地分离低秩  $n$  阶张量数据与稀疏部分。

(2)利用张量可以表征交通视频图像,可以采用张量恢复方法来提取交通视频中的交通场景(背景)与车辆目标(前景)。

(3)在交通图像背景分析中,与帧差法相比,采用 APG-TR 方法提高了运动目标车辆检测的准确性,并且具有很好的收敛速度、良好的实时性与鲁棒性。

(4)在未来的工作中,将探讨在此算法中如何进行参数的自动选择,同时,也将研究该算法在机器视觉与图像处理方面的其他应用。

#### 参 考 文 献 :

#### References :

- [1] 简林莎,段宗涛,周兴社. 智能运输系统信息平台[J]. 长安大学学报:自然科学版,2006,26(2):81-83.  
JIAN Lin-sha, DUAN Zong-tao, ZHOU Xing-she. Information platform of intelligent transportation systems [J]. Journal of Chang'an University: Natural Science Edition, 2006, 26(2): 81-83. (in Chinese)
- [2] 金 凌,陈启美,王 炜. ITS 域跨平台应用技术[J]. 交通运输工程学报,2008,8(3):101-107.  
JIN Ling, CHEN Qi-mei, WANG Wei. Cross-platform application technologies in ITS[J]. Journal of Traffic and Transportation Engineering, 2008, 8(3): 101-107. (in Chinese)
- [3] 李 毅,孙正兴,远 博,等. 一种改进的帧差和背景减相结合的运动检测方法[J]. 中国图象图形学报,2009,14(6):1162-1168.  
LI Yi, SUN Zheng-xing, YUAN Bo, et al. An improved method for motion detection by frame difference and background subtraction[J]. Journal of Image and Graphics, 2009, 14(6): 1162-1168. (in Chinese)
- [4] 高 韬,刘正光,岳士宏,等. 用于智能交通的运动车辆跟踪算法[J]. 中国公路学报,2010,23(3):89-94.  
GAO Tao, LIU Zheng-guang, YUE Shi-hong, et al. Moving vehicle tracking algorithm used for intelligent traffic[J]. China Journal of Highway and Transport, 2010, 23(3): 89-94. (in Chinese)
- [5] 冀振元,李晨雷,唐文彦. 支持向量机在车辆目标识别中的

应用[J]. 系统工程与电子技术,2010,32(2):284-286.

- Ji Zhen-yuan, LI Chen-lei, TANG Wen-yan. Application of SVM in recognition of vehicles[J]. Systems Engineering and Electronics, 2010, 32(2): 284-286. (in Chinese)
- [6] 李喜来,李艾华,白向峰. 智能交通系统运动车辆的光流法检测[J]. 光电技术应用,2010,25(2):75-78.  
LI Xi-lai, LI Ai-hua, BAI Xiang-feng. Moving vehicles detection in intelligent transportation systems based on optical flow[J]. Electro-Optic Technology Application, 2010, 25(2): 75-78. (in Chinese)
- [7] SHEN Z, TOH K C, YUN S. An accelerated proximal gradient algorithm for frame-based image restoration via the balanced approach[J]. SIAM Journal on Imaging Sciences, 2011, 4(2): 573-596.
- [8] WRIGHT J, GANESH A, RAO S, et al. Robust principal component analysis: exact recovery of corrupted low-rank matrices via convex optimization[C]//BENGIO Y, SCHUURMANS D, LAFFERTY J, et al. Advances in Neural Information Processing Systems. Cambridge: MIT Press, 2009: 2080-2088.
- [9] CANDES E J, LI Xiao-dong, MA Yi, et al. Robust principal component analysis[J]. Journal of the ACM, 2011, 58(3): 1-39.
- [10] LI Yin, YAN Jun-chi, ZHOU Yue, et al. Optimum subspace learning and error correction for tensors[C]//DANILLIDIS K, MARAGOS P, PARAGIOS N. Proceedings of the 11th European Conference on Computer Vision. Crete: IEEE, 2010: 790-803.
- [11] TSENG P, YUN S. Block-coordinate gradient descent method for linearly constrained nonsmooth separable optimization[J]. Journal of Optimization Theory and Applications, 2009, 140(3): 513-535.
- [12] ZHOU Zi-han, LI Xiao-dong, WRIGHT J, et al. Stable principal component pursuit [C] // IEEE. International Symposium on Information Theory. Texas: IEEE, 2010: 1518-1522.
- [13] CANDES E J, RECHT B. Exact matrix completion via convex optimization [J]. Foundations of Computational Mathematics, 2009, 9(6): 717-772.
- [14] TSENG P. On accelerated proximal gradient methods for convex-concave optimization[R]. Washington DC: University of Washington, 2008.
- [15] BECK A, TEBOULLE M. A fast iterative shrinkage-thresholding algorithm for linear inverse problems[J]. SIAM Journal on Imaging Sciences, 2009, 2(1): 183-202.
- [16] TSENG P. Convergence of a block coordinate descent method for nondifferentiable minimization[J]. Journal of Optimization Theory and Applications, 2001, 109(3): 475-494.
- [17] HALE E T, YIN Wo-tao, ZHANG Yin. Fixed-point continuation for  $l_1$ -minimization: methodology and convergence[J]. SIAM Journal on Optimization, 2008, 19(3): 1107-1130.
- [18] CAI Jian-feng, CANDES E J, SHEN Zuo-wei. A singular value thresholding algorithm for matrix completion [J]. SIAM Journal on Optimization, 2008, 20(4): 1956-1982.