

# 基于控制函数估计图像噪音的标准方差

王文远

(北京应用物理与计算数学研究所 北京 100094)

**摘要** 提出了一种新的估计自然图像噪音的标准方差的算法。该算法通过控制函数在经过网格化后的图像上选择某些区域,然后用一种梯度滤波算法去估计噪音的标准方差。控制函数是一个与图像的预信噪比相关的量,可以用总体误差的最小化来优化确定。控制函数可以有效地平衡图像结构对噪音估计的影响,因而不像已有的算法只能对图像有限范围的噪音强度进行有效估计。所提算法能够对从极小到极大范围的噪音强度进行准确估计。同时,整个噪音估计算法可以通过一个快速收敛的迭代算法来计算,以期获得更稳健的解。量化实验表明,相对于已有的算法,新算法估计图像噪音的标准方差更准确。

**关键词** 标准方差,网格化,控制函数,预信噪比

中图分类号 TP391.41 文献标识码 A

## Estimating the Standard Deviation of Noise via Controlled Function

WANG Wen-yuan

(Institute of Applied Physics and Computational Mathematics, Beijing 100094, China)

**Abstract** This paper presented a new algorithm to estimate the standard deviation of noise for nature images. It uses a gradient filter to estimate the noise on some parts which are selected by a controlled function from the tessellating image. The controlled function can be related to the preestimate of SNR. We optimally determined the controlled function as an exponential function by minimizing the total error. The controlled function can effectively offset the effect which is caused by filtering the image structure. Therefore unlike some existing algorithms which can only accurately estimate noise for images with limited range of noise intensity, the proposed algorithm can perfectly do this for images with the noise intensity range of extremely small to extremely heavy. Moreover, an iterative process which has fast convergence was implemented to obtain more robust solution. Quantitative comparisons with some existing algorithms on an image data set demonstrate that the proposed algorithm has high accuracy, and outperforms those existing algorithms used in this study.

**Keywords** Standard deviation, Tessellating, Controlled function, Preestimate of SNR

估计图像噪音的标准方差是一项重要的研究内容。准确的噪音标准方差估计是很有用的,因为图像处理中的很多算法,比如边界探测<sup>[1]</sup>、图像复原<sup>[2]</sup>、阈值选取<sup>[3]</sup>、图像分割<sup>[4]</sup>、非线性扩散滤波<sup>[5]</sup>和图像重建<sup>[6]</sup>都将噪音标准方差作为一个参数。不同成像模式的图像噪音有着相应的标准方差估计算法,这表现在远程声纳图像<sup>[7]</sup>、扫描电子显微镜图像<sup>[8]</sup>、核磁共振图像<sup>[9]</sup>、语音信号<sup>[10]</sup>、视频图像<sup>[11]</sup>和天文图像<sup>[12]</sup>等方面。噪音标准方差估计难度在于图像信息和噪音混合在一起,图像结构对噪音估计有影响。对于自然图像中的噪音(通常可近似为高斯噪音)估计,常用的算法是用原图像减去低通滤波后的图像再计算标准方差。近年来此方面的研究相对较少<sup>[13-15]</sup>,最近由 Rank 等提出的算法首先通过两个串联的一维差分算子对图像进行滤波,以抑制图像结构;其次计算图像局域标准方差的直方图;最后通过评估该直方图以得到噪音标准方差的估计。

本文针对自然图像的高斯噪音,提出了一种新的图像噪音标准方差估计算法,用以提高噪音标准方差估计的准确度。首先比较了多种低通滤波算法,选取了一个梯度算法应用在本研究中。其次将滤波后的图像网格化,并调查所用的网格数和准确估计噪音标准方差之间的关系,可发现该关系可用一个关于图像信噪比的预估计的控制函数来表示。虽然每一幅图像都有其各自相适应的控制函数,但为了寻找对一组图像适用的控制函数,本文采用了训练集和最优化算法来确定:先通过在少量样本图像上的计算,评估出一个带双参数的指数函数作为控制函数,然后在更多的训练图像中确定双参数值。另外,整个算法可迭代运算,以期获得更稳定的解。

## 1 算法

定义  $x(m, n)$  是一个无噪音的二维灰度图像。如果该图像有  $M \times N$  个像素点,可定义区域  $D_0 = \{(m, n) | m \in [1, M]$

到稿日期:2008-10-31 返修日期:2009-02-09 本文受国家自然科学基金(10576013),中国工程物理研究院科学发展基金重点项目(2007A01001)资助。

王文远(1972-),男,副研究员,主要研究方向为图像处理、模式识别和放射线照相物理, E-mail: wang\_wenyuan@iapcm.ac.cn.

$\wedge n \in [1, N]\}$ 。同样的方法可定义一高斯噪声退化图像  $y(m, n)$ ;

$$y(m, n) = x(m, n) + w(m, n) \quad \forall (m, n) \in D_0 \quad (1)$$

高斯噪声  $w(m, n)$  是零均值的可加性噪声, 被假定在统计上独立于原始图像  $x$ 。原始图像  $x$  和噪声  $w$  在统计意义上的混合是准确估计噪声的主要困难。由式(1),  $y, x$  和  $w$  的标准方差  $\sigma_y, \sigma_x$  和  $\sigma_w$  的关系式可以表示为

$$\sigma_y^2 = \sigma_x^2 + \sigma_w^2 \quad (2)$$

直接从噪声退化图像  $y$  中计算标准方差作为噪声的估计是不行的, 因为噪声和图像结构都可贡献于其中。而这两种信号的分离是很困难的。解决该问题的常用方法是用一些滤波来抑制原始图像的结构, 然后再估计噪声。常用的滤波有高斯滤波、中值滤波和梯度滤波。在这 3 种滤波中, 相比于高斯滤波和中值滤波存在需要调节的滤波尺度, 梯度滤波无调节参数, 且能更好地抑制图像结构, 所以在本研究中选取梯度滤波作为抑制图像结构的算法。定义梯度滤波为

$$H_{\text{grad}} = \frac{1}{2} \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix} \quad (3)$$

函数  $y$  经梯度滤波后的标准方差记为

$$\hat{\sigma}_y = \text{std}(H_{\text{grad}} * y) \quad (4)$$

其中,  $\text{std}$  表示标准方差的计算, 此即为经梯度滤波后估计噪声标准方差的算法。

经过梯度滤波后的图像

$$Y = H_{\text{grad}} * y \quad (5)$$

可网格化成尺寸为  $\ell \times \ell$  的正方形小块, 这些小块标号为  $T_j$ ,  $j \in [1, \omega]$ ,  $\omega$  应该等于或接近  $mn/(\ell\ell)$ 。

所有这些小块的标准方差按从小到大重新排序为  $s_j$ ,  $j \in [1, \omega]$ , 与  $s_j$  相对应的小块定义为  $T_{s_j}$ 。给定如下序列区域

$$Y_J = [T_{s_1}, T_{s_2}, \dots, T_{s_J}] \quad J \in [1, \omega] \quad (6)$$

定义  $Y_J$  的标准方差为  $s_J$ , 可期望  $J$  越大, 则  $s_J$  越大。所以控制  $J$  可获得相应的  $s_J$  接近噪声的真实值。定义  $J_c$  作为控制  $J$  而得到噪声的准确估计的解。那么, 如果得到  $J_c$ , 即可通过

$$\hat{\sigma}_w = \text{std}(Y_{J_c}) = \text{std}([T_{s_1}, T_{s_2}, \dots, T_{s_{J_c}}]) \quad (7)$$

得到噪声标准方差的准确估计。

通过在一些真实图像上加不同程度的噪声进行分析, 可得到  $J_c$  和噪声的准确估计之间的关系。该关系可通过一个相关于预信噪比的控制函数来描述。预信噪比可以用来预先判断图像受噪声退化的程度。定义预信噪比的经验公式如下:

$$\text{SNR}_p = \frac{I_s}{\sigma_p} \quad (8)$$

其中,  $I_s$  是给定图像信号强度的估计;  $\sigma_p$  则是给定图像噪声标准方差的预估计, 可以通过式(4)和式(3)得到, 即令

$$\hat{\sigma}_p = \hat{\sigma}_y = \text{std}(H_{\text{grad}} * y) \quad (9)$$

作为图像噪声标准方差的预估计。定义  $I_s$  的计算式为

$$I_s = \frac{\max(y * G_\alpha) - \min(y * G_\alpha)}{2} \quad (10)$$

这里近似地把信号中值作为信号强度的估计。带大尺度  $\alpha$  (本研究中取  $\alpha=6$ ) 的高斯滤波  $G_\alpha$  能较好地清除不同噪声对  $I_s$  估计的影响, 可使得在同一幅图像中  $I_s$  估计随着噪声的增加保持基本不变。

如何确定控制函数而得到  $J_c$ , 是本研究中的重要问题。

每幅图像都有其通过强度、背景和纹理所描述的独一无二的特性, 因此都有各自相应的控制函数。对拥有大量图像的数据库建立一个通用的控制函数是非常困难的, 实现此目标的唯一办法是在不同图像上进行统计意义上的折衷, 以期获得一个有实践意义的控制函数。它对于拥有大量图像的数据库, 能最大程度地描述  $J_c$  和图像预信噪比之间的关系。

寻找控制函数可分为两步: 第一步为减少计算量, 用少数几个训练图像(但需要包括图像的各种特性)计算各自相对于不同的  $\text{SNR}_p$  的  $J_c$ 。从图像库<sup>[16]</sup>中选取 5 幅图像, 包括大背景景观、纹理丰富、肖像、建筑物和动物图像各一幅。标准方差  $\sigma_w$  从 3 到 123 之间的 21 个逐渐增大的高斯噪声附加在这 5 幅图像上。对这  $21 \times 5$  幅噪声图像计算  $J_c$ 。图 1 中带星号的曲线描述了  $J_c/\omega$  相对于  $\log(\text{SNR}_p)$  的关系。从这些曲线图中可给出一个带双参数的指数函数作为控制函数。

$$J_c/\omega = c_1 \exp(-\text{SNR}_p/c_2) \quad c_1 > 0, c_2 > 0 \quad (11)$$

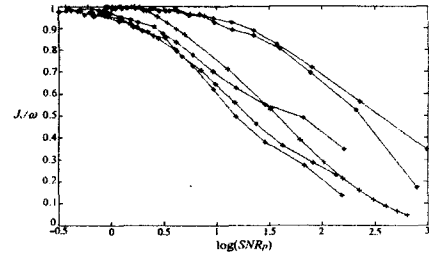


图 1 在 5 个训练图像上  $J_c/\omega$  相对于  $\log(\text{SNR}_p)$  的曲线图(带星号)和控制函数(带加号)

然而, 为了应用更准确, 确定  $c_1, c_2$  需要在更多的训练图像上计算。在给定的训练图像是无噪声图像的情况下, 可以用最优化准则来确定  $c_1, c_2$ 。对于每一对  $c_1, c_2$ , 都可得到每个图像  $x_i$ 、每个附加噪声  $\sigma_w$  相对应所计算出的  $\hat{\sigma}_w$ 。于是相对于  $c_1, c_2$  的总的误差为

$$E(c_1, c_2) = \sum_{\sigma_w} \sum_{x_i} |\hat{\sigma}_w - \sigma_w| \quad (12)$$

故第二步, 从文献<sup>[16]</sup>的图像库上随机地选取 30 幅图像作为训练集, 标准方差  $\sigma_w$  从 3 到 123 之间的 21 个逐渐增大的高斯噪声附加在这 30 幅图像上, 总共生成  $21 \times 30$  幅噪声图像。变化  $c_1$  和  $c_2$ , 用式(10)、式(11)和式(7)对每一幅噪声图像计算  $\hat{\sigma}_w$ 。利用已知的  $\sigma_w$  最小化式(12), 可得到  $c_1 = 1.5$ ,  $c_2 = 5.0$ , 这就是得到的一组优化参数。图 1 中带加号的曲线描述了式(11)在  $c_1 = 1.5, c_2 = 5.0$  时  $J_c/\omega$  相对于  $\log(\text{SNR}_p)$  的关系。

最后, 将  $\sigma_p$  作为噪声标准方差估计的初始值, 则整个算法可迭代运算, 以期获得更稳定的解。

新的噪声估计算法可总结为:

- (1) 用  $H_{\text{grad}}$  滤波原始图像  $y$  得到  $Y = H_{\text{grad}} * y$ , 计算  $\sigma_p = \text{std}(Y)$ , 令  $\hat{\sigma}_w^1 = \sigma_p$ ;
- (2) 网格化  $Y$  为  $\ell \times \ell$  个正方块(本研究中  $\ell=21$ );
- (3) 从(4)至(6)迭代运算  $it=1, 2, \dots$ , 直到收敛;
- (4) 通过  $\text{SNR}_p^{it+1} = \frac{I_s}{\hat{\sigma}_w^{it+1}}$  计算  $\text{SNR}_p^{it+1}$ ;
- (5) 通过

$$J_c^{it+1}/\omega = \begin{cases} 1.4 \exp(-\text{SNR}_p^{it+1}/5.0), & \text{If } \text{SNR}_p^{it+1} > 2.253 \\ 1, & \text{If } \text{SNR}_p^{it+1} \leq 2.253 \end{cases}$$

(注意  $J_c/\omega$  应小于或等于 1) 计算  $J_c^{it+1}$ ;

(6)通过 $\hat{\sigma}_w^{i+1} = \text{std}(Y_{j_p}^{i+1})$ 计算 $\hat{\sigma}_w^{i+1}$ 。

本算法可通过相应的导数分析证明其收敛性。接下来的实验将演示其收敛速度非常快。

## 2 实验

实验1 调查新算法的迭代收敛性。图2(a)为一样本图像,标准方差 $\sigma_a$ 为9和63的高斯噪声附加在这幅图像上。运用新算法,相应的计算结果分别如图2(b)和(c)所示。在其中选取 $it=6$ 和3个初始值 $\hat{\sigma}_w^1=1$ , $\hat{\sigma}_w^1=\sigma_p$ 和 $\hat{\sigma}_w^1=150$ (分别对应于图中带加号、带星号和带交叉号的曲线)。可见新算法收敛速度非常快,即便对于非常大或非常小的初始值也只需要3步迭代运算即可收敛。在本研究接下来的实验中,取 $\hat{\sigma}_w^1=\sigma_p$ 和 $it=2$ 来进行计算。

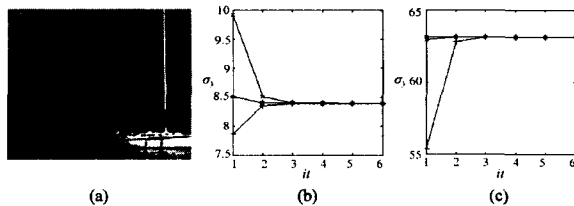


图2 调查新算法的收敛性

实验2 在文献[16]图像库中的300个图像上调查新算法的准确性。选取4个已有的算法来比较,包括用式(4)和式(3)来估计的梯度滤波算法,如Olsen<sup>[15]</sup>讨论的优于其它算法的平均滤波算法、Rank的算法<sup>[14]</sup>和Meer的算法<sup>[13]</sup>。标准方差 $\sigma_a$ 从3到123之间的21个逐渐增大的高斯噪声附加在这300幅图像上。对于这 $300 \times 21$ 个噪声图像用包括新算法在内的5种算法来估计噪声的标准方差 $\hat{\sigma}_w$ 。

误差的绝对值

$$\epsilon(i, \sigma_a) = |\hat{\sigma}_w - \sigma_a| \quad (13)$$

(这里 $i \in [1, 300]$ 定义为图像的序列数), $\epsilon$ 的平均值

$$\bar{\epsilon}(\sigma_a) = \frac{1}{300} \sum_i \epsilon(i, \sigma_a) \quad (14)$$

$$\bar{\epsilon} = \frac{1}{300 \times 21} \sum_i \sum_{\sigma_a} \epsilon(i, \sigma_a)$$

和标准方差

$$\sigma_\epsilon^2 = \frac{1}{300 \times 21} \sum_i \sum_{\sigma_a} (\epsilon(i, \sigma_a) - \bar{\epsilon})^2 \quad (15)$$

用来进行分析。

图3显示的是 $\bar{\epsilon}$ 相对于 $\sigma_a$ 的曲线图,其中带星号、点、圆圈、交叉号和加号的曲线分别对应于新算法、梯度算法、平均值算法、Rank算法和Meer算法。可看到新算法的估计结果最好。进一步的对比分析为:梯度算法在 $\sigma_a$ 大于51时给出的结果和新算法接近,但当 $\sigma_a$ 较小时其估计值的误差较大;平均值算法给出和梯度算法相同趋势的误差曲线,但其误差更大;Rank算法仅仅在 $\sigma_a \in (20, 40)$ 给出较好的结果;Meer算法在绝大多数情况下平均误差大于2;对所有的 $\sigma_a$ ,新算法的平均误差小于0.7。

表1列出了一些关于估值误差的统计值。可看到新算法估计值的总的平均值和标准方差仅仅分别为0.421和0.905,且在所有 $300 \times 21$ 次计算中,新算法误差小于0.5和1的分别占76.2%和93.2%。这些结果表明新算法在量化对比中优于其它算法。而在计算效率上,由于新算法和Rank的算

法都采用了迭代算法,因而它们相应的计算时间相对于简单的梯度和平均值算法要长。

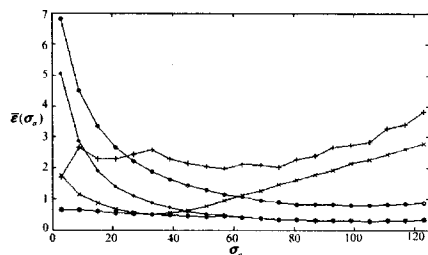


图3 从 $\bar{\epsilon}$ 相对于 $\sigma_a$ 的曲线图上对比各种算法的计算准确性

表1 从计算误差的总平均值、标准方差和误差小于某值的成功率来比较各种算法的计算准确性

| 方法                | 新算法   | 梯度    | 平均值   | Rank  | Meer  |
|-------------------|-------|-------|-------|-------|-------|
| $\bar{\epsilon}$  | 0.421 | 0.905 | 1.737 | 1.421 | 2.484 |
| $\sigma_\epsilon$ | 0.618 | 1.517 | 2.665 | 1.022 | 2.616 |
| $\epsilon < 0.5$  | 76.2% | 59.2% | 34.0% | 21.4% | 14.7% |
| $\epsilon < 1$    | 93.2% | 77.2% | 56.2% | 39.4% | 29.7% |
| $\epsilon < 5$    | 99.7% | 96.7% | 91.8% | 99.4% | 89.5% |
| $\epsilon < 10$   | 100%  | 99.6% | 97.7% | 100%  | 98.4% |
| T(s)              | 1.12  | 0.12  | 0.15  | 1.15  | 0.65  |

T表示各算法每次计算的平均时间(标准C,奔腾2.0CPU)

**结束语** 本文提出的算法对梯度滤波后的图像进行网格化,然后经控制函数选取某些区域进行噪声的标准方差估计。实验中可见利用总体误差的最小化求得的控制函数较好地解决了不同强度噪声的准确估计问题,在实际应用中有一定的意义。

## 参考文献

- [1] Jiang L H, Guo Y Y. Image degedetection based on adaptive weighted morphology[J]. Chin. Opt. Lett., 2007, 5: 77-78
- [2] Mohsen E M, Mansour J. Linear motion blur parameter estimation in noisy images using fuzzy sets and power spectrum[J]. EURASIP Journal on Advances in Signal Processing, 2007; 1-8
- [3] Sahoo P K, Soltani S, Wong A K C. A survey of thresholding techniques[J]. Comput. Vision, Graphics, Image Processing, 1988, 41: 233-260
- [4] Besl P J, Jain R C. Segmentation through variable-order surface fitting[J]. IEEE Trans. Pattern Anal. Machine Intell., 1988, 10: 167-192
- [5] Lysaker M, Lundervold A, Tai X C. Noise removal using fourth-order partial differential equations with applications to medical magnetic resonance images in space and time[J]. IEEE Trans. Image Proc., 2007, 12: 1579-1590
- [6] Fessler J A. Mean and variance of implicitly defined biased estimators (such as penalized maximum likelihood): Applications to tomography[J]. IEEE Trans. Image Proc., 1996, 5: 493-506
- [7] Corner B R, Narayanan R M, Reichenbach S E. Noise estimation in remote sensing imagery using data masking[J]. Int. J. Remote Sensing, 2003, 24: 689-702
- [8] Thong J T L, Sim K S, Phang J C H. Single-image signal-to-noise ratio estimation[J]. Scanning, 2001, 23: 328-336
- [9] Sijbers J, den Dekker A J. Maximum likelihood estimation of signal amplitude and noise variance from MR data[J]. Magnetic Resonance in Medicine, 2004, 51: 586-594
- [10] Park Y S, Chang J H. A probabilistic combination method of

minimum statistics and soft decision for robust noise power estimation in speech enhancement[J]. IEEE Signal Processing Letters, 2004, 11: 95-98

- [11] Amrt A, Mitiche A, Dubois E. Reliable and fast structure-oriented video noise estimation[J]. IEEE Proc. Int. Conf. on Image Processing, 2002, 1: 840-843
- [12] Starck J L, Murtagh F. Automatic noise estimation from the multiresolution support[J]. Publications of the Astronomical Society of the Pacific, 1998, 110: 193-199
- [13] Meer P, Jolion J M, Rosenfeld A. A fast parallel algorithm for

blind estimation of noise variance[J]. IEEE Trans. Pattern Anal. Machine Intell., 1990, 12: 216-223

- [14] Rank K, Lendl M, Unbehauen R. Estimation of image noise variance[J]. IEE Proc. Vis. Image Signal Process, 1999, 146: 80-84
- [15] Olsen S L. Estimation of noise images: An evaluation[J]. CV-GIP. Graph models Image Process., 1993, 55: 319-323
- [16] Martin D R, Fowlkes C C, Malik J. Learning to detect natural image boundaries using local brightness, color, and texture cues[J]. IEEE Trans. Pattern Anal. Machine Intell., 2004, 26: 530-549

(上接第 257 页)

存替换机制的算法时间复杂度为  $O(k \log k)$ 。每个 peer 节点的缓存空间容量是有限的,且流媒体片段通常会以  $M$  为基本单位。因此  $k$  是一个不大的数字,相对于目前计算机的处理速度而言,时间复杂度  $O(k \log k)$  所需要的时间是较少的。

## 5 性能分析

系统拓扑采用构造工具 GT-ITM<sup>[8]</sup>生成 transit-stub 层次模型。系统生成 3 个 transit 域,transit 域的平均骨干节点数为 4,骨干节点平均连接 2 个 stub 域。每个 stub 域的平均 host 节点数为 3 个。系统中每个 host 节点均为一个 peer 节点,系统节点数为 72 个。在 transit 域中选择一个骨干节点作为流媒体服务器,放置 100 个媒体文件,媒体文件的数据容量为 150M~1G,编码率为 300~500kbps。在 P2P 流媒体系统中,对流媒体片段的访问服从 Zipf 分布,模拟环境中,假设对流媒体片段的访问服从  $\theta=0.20$  的 Zipf 分布。节点对媒体文件的访问请求以一个恒定的速率到达。假设节点以恒定的速率加入和离开系统,测试节点缓存大小为 10M~200M 变化下的系统性能。在系统性能稳定时进行取值,节点缓存大小变化时的流媒体片段副本数、服务器连接数(反映服务器负载状况)、启动延时的变化如图 2、图 3、图 4 所示。

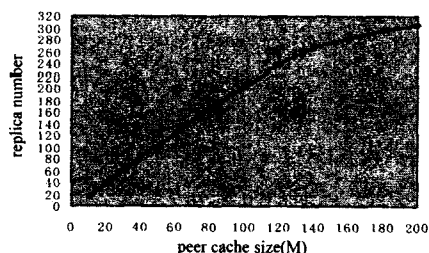


图 2 缓存大小不同的副本数

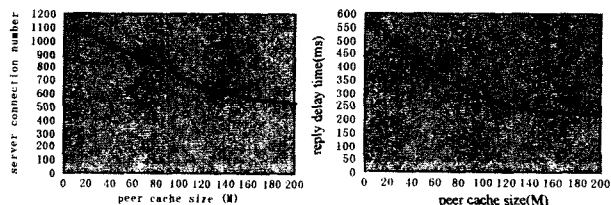


图 3 不同缓存大小时的服务器连接数

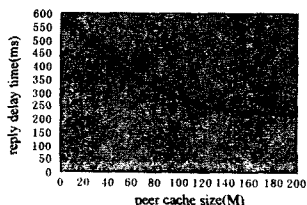


图 4 不同缓存大小时的请求响应延迟

图 2 显示的是 peer 节点数恒定情况下,peer 节点缓存大小从 10M~200M 变化时的系统流媒体片段副本数状况。从图 2 可以看出,随着 peer 节点缓存空间的加大,流媒体片段的副本数不断增加。

图 3 显示的是 peer 节点缓存大小变化时服务器连接数的情况。该图表明节点缓存空间加大,服务器的连接负载情况就会明显减少。在缓存从 10M 变化至 120M 时,服务器端的连接数显著降低,之后降低速度逐渐放缓。

图 4 显示的是 peer 节点缓存大小变化时服务请求响应延迟的状况。该图说明节点缓存空间加大,流媒体视频播放响应延迟则缩短。在缓存从 10M 变化至 100M 时,响应延迟缩短速度快,之后缩短逐步缓慢。

综合解读图 2、图 3、图 4 可看到,随着系统整体缓存空间的加大,流媒体片段副本数随之增加,系统的整体响应时间延迟缩短,骨干网络通信负载降低。表明系统缓存空间大小对流媒体服务质量有较大的影响,因此,有效利用缓存空间可以有效地改善流媒体的服务质量。

**结束语** 本文提出了一种对等网络混合式拓扑结构下的 CORPC 缓存管理方案。该方案在考虑流媒体片段流行度的同时,最大限度地利用了节点缓存空间。通过缓存空间分配算法,兼顾了流行度较高和较低的流媒体片段的服务质量。将来的工作包括进一步完善该策略,在缓存预取等方面做进一步研究。

## 参考文献

- [1] <http://www.gnutella.com>, Gnutella website
- [2] <http://www.kazaa.com>, KaZaA website
- [3] Saroiu S, Gummadi P, Gribble S. A measurement study of peer-to-peer file sharing systems[C]//Proceedings of MMCN02. San Jose, CA, USA, January 2002
- [4] Hefeeda M M, Bhargava B K, Yau D K Y. A hybrid architecture for cost-effective on-demand media streaming[J]. Computer Networks, 2004, 44(3)
- [5] Guo, Chen L S, Zhang X. Design and evaluation of a scalable and reliable P2P assisted proxy for on demand streaming media delivery[J]. IEEE Transactions on Knowledge and Data Engineering, 2006, 18(5): 669-682
- [6] Ying L, Basu A. pcVOD: Internet Peer-to-Peer Video-On-Demand with Storage Caching on Peers[C]//Proceedings of DMS, 2005
- [7] Cheng Bin, Stein L, Jin H, et al. Towards cinematic internet video-on-demand[C]//EuroSys. 2008: 109-122
- [8] Calvert K, Zegura E. GT2ITM: Georgia Tech Internetwork Topology Models[OL]. <http://www.cc.gatech.edu/fac/Ellen.Zegura/graphs.html>, 2006-10