

# Comparison of Deinterlacing Filters

28 September 2003

## Table of contents

Table of contents .....	1
Introduction .....	1
Описание работы и сравнение параметров фильтров .....	2
VirtualDub Internal .....	2
Area Based v1.4 by Gunnar Thalin .....	3
Smart Deinterlace v2.7 by Donald Graft .....	6
Smooth v1.1 by Gunnar Thalin .....	10
Smart Bob v1.1 by Gunnar Thalin .....	11
Сравнение существующих фильтров для деинтерлейсинга на различных тестовых последовательностях .....	11
bbc.avi frame number 13 .....	12
susi.avi frame number 14 .....	14
tennis.avi frame number 18 .....	16

## Introduction

В этом обзоре рассмотрены некоторые из наиболее распространенных фильтров для деинтерлейсинга. Стоит отметить, что все эти фильтры не используют механизм компенсации движения, который сложен и трудоёмок, но позволяет значительно повысить качество деинтерлейсинга. К сожалению, свободно доступных фильтров, использующих механизм компенсации движения, по-видимому, нет.

Все фильтры разработаны для программы VirtualDub.

Итак, в статье рассмотрены следующие фильтры:

- VirtualDub Internal filter
- Area Based v1.4 filter
- Smart Deinterlace v2.7 filter
- Smooth v1.1 filter
- Smart Bob v1.1 filter

Для этих фильтров приведено краткое описание принципа работы и сравнение работы фильтра при различных параметрах. И, в заключение, результаты обработки этими фильтрами при найденных нами «лучших» (понятно, что панацеи не бывает, поскольку подбор параметров для фильтров определяется спецификой данного фильма) параметрах ряда фильмов.

**Описание работы и сравнение параметров фильтров****VirtualDub Internal**

Самый простой фильтр из всех рассмотренных в статье. Чтобы получить на выходе картинку того же размера, что и на входе, придется включить либо режим **Duplicate Field**, либо более качественный **Blend Fields Together**

В режиме **Duplicate Field** каждая чётная строка заменяется предыдущей нечётно или наоборот. В результате теряется половина информации о кадре, и изображение получается сильно ступенчатым.

В режиме **Blend Fields Together** производится размытие фрейма. Результат более качественный, однако, изображение сильно размытое при интенсивном движении.

Стоит сказать, что кадр обрабатывается целиком, не зависимо от того было движение или нет.

**Результат работы фильтра при различных параметрах:**



Picture 1. Source image (bbc.avi; 10 frame)



Picture 2. Duplicate Field



Picture 3. Blend Fields Together



Picture 4. Source image (mobl.avi; 261 frame)



Picture 5. Duplicate Field



Picture 6. Blend Fields Together

### Area Based v1.4 by Gunnar Thalin

Более сложный метод, учитывающий наличие или отсутствие интерлэйсинга. Точнее, фильтр ищет в каждом кадре interlaced-линии, а когда он находит их, удаляет либо путем интерполяции, либо блендинга (режим блендинга даёт те же артефакты что в предыдущем случае, поэтому в примерах используется режим интерполяции). Фильтр не интересуется, откуда появился интерлэйсинг (движение объектов, или же результат работы MPEG-кодека со сжатыми 2-field файлами). Хорошо подходит там, где только часть кадра (или же лишь отдельные кадры) подвержены интерлэйсингу. Дает в целом неплохой результат с хорошей четкостью там, где нет движения. Недостаток - характерные для интерлэйсинга артефакты не исчезают полностью даже при уменьшении порогов обнаружения движения и границ до нуля (поэтому, для лучшей обработки интерлэйсинга будем использовать **threshold=0**).

Результат работы фильтра при различных параметрах:



Picture 7. Source image (mobl.avi; 261 frame)



Picture 8. edge=15



Picture 9. edge=20



Picture 10. edge=25



Picture 11. edge=50



**Picture 12. Source image (mobl.avi; 261 frame)**



**Picture 13. edge=15**



**Picture 14. edge=20**



**Picture 15. edge=25**



**Picture 16. edge=50**

Видим, что «золотой серединой» являются настройки: **threshold=0, edge=25.**

## Smart Deinterlace v2.7 by Donald Graft

Один из лучших фильтров для VirtualDub на текущий момент.

Алгоритм работы довольно прост:

- строит карту движения
- если выбрано - производит шумоподавление карты с помощью морфологической операции закрытия
- интерполирует либо линейно, либо, если выбрано, кубически (с коэффициентами -1, 5, 5, -1) пиксели, помеченные как движущиеся
  - может выполнять деинтерлейсинг размытием, но этот метод уже обсуждался

Как следствие, при правильном подборе **motion threshold** получаем нетронутые области там, где движения не было, и малый процент артефактов там, где движение было.

Карта движения может строиться тремя способами:

- сравнивая текущий фрейм с предыдущим (**Frame-Only Differencing**)
- сравнивая поля одного фрейма между собой (**Field-Only Differencing**)
- либо совмещая и то, и другое (**Frame-and- Field Differencing**)

Сравнение может проходить, как по трем компонентам цвета, так и по освещенности. В качестве порога используется **motion threshold**.

Главная проблема - подбор оптимального порога, зависящего от конкретной последовательности. Кроме того, если у нас сильно различается интенсивность движения в пределах кадра, то эта проблема вообще корректно не решается. Ещё один недостаток - нет механизма сохранения границ, как в **Area Based**.

### Результат работы фильтра при различных параметрах:

Сначала сравним разные методы построения карт движения, поскольку данный момент является определяющим. Будем использовать **Compare Color Channels** и **Motion Map Denoising, motion threshold = 10**. Интерполяция – бикубическая.



Picture 17. Source image (tennis.avi; 2 frame)

Далее приведены результаты работы фильтра при различных значениях **Differencing**: слева приведены карты движения, а справа результат обработки изображения при этой карте.



**Picture 18. Field Only**



**Picture 19. Processed Images**



**Picture 20. Frame-Only**



**Picture 21. Processed Images**



**Picture 22. Frame-and-Field**



**Picture 23. Processed Images**

Видим, что лучший результат даёт установка **Frame-Only Differencing**.  
Итак, далее используются настройки: **Frame-Only Differencing, Compare Color Channels, Motion Map Denoising** и **Cubic Intrpolation**.

Результат работы фильтра при различных параметрах:

Первое сравнение проведём на том же кадре, что и для карты движения, т.е. **tennis.avi, frame 2.**



**Picture 24. threshold = 5**



**Picture 25. Processed Images**



**Picture 26. threshold =10**



**Picture 27. Processed Images**



**Picture 28. threshold =20**



**Picture 29. Processed Images**



Другой фрагмент из этого кадра:

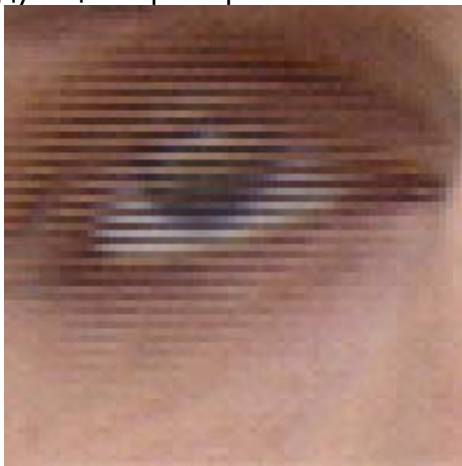


Picture 30. *threshold = 5*



Picture 31. *threshold = 10*

Следующий пример **susi.avi frame 14**:



Picture 32. Source image



Picture 33. *threshold = 5*



Picture 34. *threshold = 10*



Picture 35. *threshold = 20*

Стоит отметить, что при ***threshold = 5*** фильтр не только убирает интерлейсинг, но и портит некоторые детали, а так же большую часть шумов (поскольку интерполирует почти по всему кадру). То, что фильтр убирает шумы не очень хорошо, так как он делает это не качественно, а есть фильтры специально для борьбы с шумами.

При правильном подборе порога (например, ***threshold = 10***) фильтр довольно качественно выполняет обработку.

**Smooth v1.1** by Gunnar Thalin

Данный фильтр выдаёт на выходе видео с удвоенной частотой кадров. Для этого требуется предварительно разделить входную последовательность фреймов на последовательность полей, соответственно с удвоенной частотой. Это можно сделать с помощью утилиты AVISynth.

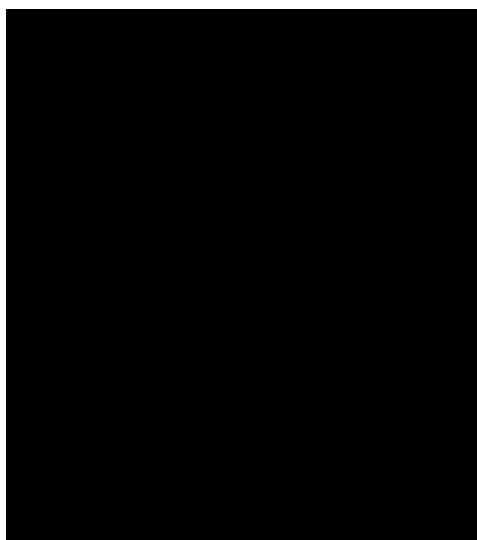
Разработчик **Smooth** заявляет, что этот фильтр работает так же, как **Area Based**, только выдает удвоенную частоту кадров, что и было замечено на опыте:



**Picture 36. Source image**



**Picture 37. Smooth**



**Picture 38. LUV metrics**



**Picture 39. Area Based**

## **Smart Bob v1.1** by Gunnar Thalin

Данный фильтр выдаёт на выходе видео с удвоенной частотой кадров. И является аналогом **Smart Deinterlace**.

### **Сравнение существующих фильтров для деинтерлейсинга на различных тестовых последовательностях**

Далее сравниваются результаты работы четырёх фильтров деинтерлейсинга:

- встроенный в VirtualDub фильтр в режиме **Duplicate Field** ( далее обозначается **Duplicate** )
- фильтр **Area Based** с параметрами: **threshold=0, edge=25** ( далее обозначается **Area Based** )
- фильтр **Smart** с параметрами: **Frame-Only Differencing, Compare Color Channels, Motion Map Denoising, Cubic Inrpolation, threshold=10** ( далее обозначается **Smart** )
- фильтр **Smart** с теми же параметрами, но в режиме **Blend** ( далее обозначается **Blend** )

Используются следующие тестовые последовательности:

- ролик `bbc.avi`
- ролик `susi.avi`
- ролик `tennis.avi`

**bbc.avi frame number 13**



**Picture 40. Source image (bbc.avi; 13 frame)**

Далее рассматривается увеличенный фрагмент из этого кадра.



**Picture 41. Фрагмент исходного изображения (bbc.avi; 13 frame)**



**Picture 42. Duplicate**



**Picture 43. Area Based**



**Picture 44. Smart**



**Picture 45. Blend**

**susi.avi frame number 14**



**Picture 46. Source image (susi.avi; 14 frame)**

Далее рассматривается увеличенный фрагмент из этого кадра.



**Picture 47.** Фрагмент исходного изображения (susi.avi; 14 frame)



**Picture 48.** Duplicate



**Picture 49.** Area Based



**Picture 50.** Smart



**Picture 51.** Blend

**tennis.avi frame number 18**



**Picture 52. Source image (tennis.avi; 18 frame)**

Далее рассматривается увеличенный фрагмент из этого кадра.





**Picture 53.** Фрагмент исходного изображения (tennis.avi; 18 frame)



**Picture 54.** Duplicate



**Picture 55.** Area Based



**Picture 56.** Smart



**Picture 57.** Blend

Ещё один фрагмент из tennis.avi frame number 18:



**Picture 58. Фрагмент исходного изображения**



**Picture 59. Duplicate**



**Picture 60. Area Based**



**Picture 61. Smart**



**Picture 62. Blend**