## Цветность пикселей, цветовые схемы, светимость точки, величина Grayness rate index – GrIx

Величина, представляющая собой характеристику степени серости / отсутствия цвета в точке

Степень отсутствия цветности – степень близости всех трех цветовых компонент друг к другу.

За величину GrIx беру стандартное отклонение R,G,B нормированное на общую светимость в точке

Где с – показатель интенсивности цвета по каждой из компонент (0:255), Y – общая светимость точки (0:255)

Здесь

clin – линейная компонента цветности в точке в пределах 0:1

cRGB = СRGB/255 – компонента цветности в точке в пределах 0:1 в схеме sRGB

CRGB – компонента цветности в точке в пределах 0:255

Для такую величину GrIx можно назвать степенью серости или степенью отсутствия цветности. Такая величина покажет, насколько сильно в каждой точке снимка отсутствует цветность.

В частности, в черных, белых или иных точках, в которых R=G=B, GrIx = 0.

Чем больше точка окрашена в отличный от серого (любой интенсивности) цвет, тем больше в последовательности (R,G,B) стандартное отклонение, тем больше величина GrIx.

Поскольку у компонент R,G,B пределы 0:255, стандартное отклонение не превышает 255.

Линейная светимость распределена от 0 до 255 и в исключительно редких случаях может быть меньше 1, что дает основания полагать, что GrIx в подавляющем большинстве случаев будет лежать в пределах от 0 до 1.

Исключение могут составлять только случаи, когда R+G+B < 3, но и в этом случае величина GrIx не превысит 1.

Это дает основание предположить, что величина GrIx может достаточно однозначно характеризовать степень отсутствия цвета в каждой точке снимка.

В свою очередь степень серости есть степень отличия цвета облака от цвета неба на снимке.

Распределение GrIx при солнечной засветке

Фильтрация солнечной засветки

Алгоритм фильтрации, предполагающий линейную зависимость убывания фоновой засвеченной обесцвеченности с расстоянием от края солнечной засветки. Заметим, что алгоритм дает свои результаты, и неплохие. При этом опорные точки для аппроксимации этой зависимости берутся следующие:

1. Край солнечной засветки: точно известно, что расстояние = 0, GrIx = 1;
2. Вторая точка определяется следующим образом
   1. Замечено, что практически на всех снимках области с величной GrIx <0.65 можно гарантированно считать областями, свободными от облаков
   2. Берется среднее расстояние до точек «неба» с учетом веса, который определяется как степень «небесности» или степень окрашенности. То есть, чем ниже GrIx, тем больше вес. Безусловно, веса нормированы таким образом, чтобы в сумме дать 1. Это среднее расстояние Rs берется за реперную точку по r (расстоянию от края солнечной засветки);
   3. На этом среднем расстоянии (это окружность) берется среднее значение GrIx с тем же весом, что использовался для определения среднего расстояния до окрашенных точек изображения. Это среднее значение берется за реперную точку по значению GrIx;
3. По полученным двум точкам аппроксимируется зависимость фоновой обесцвеченности за счет солнечной засветки.

По полученной зависимости данные GrIx модифицируются следующим образом: в для значений r от 0 до Rs вычитается значение аппроксимированной фоновой обесцвеченности, за пределами этой окружности значения не модифицируются. Сам солнечный круг в этом алгоритме подавляется до значения 0. Результат дает

Замечено, что зависимость убывания фоновой обесцвеченности с расстоянием от края солнечной засветки оказывается убедительно нелинейной:

- имеется ярко выраженный минимум

- в большинстве случаев ярко выражена нелинейность с расстоянием как до точки минимума, так и после нее

Новый алгоритм определения и компенсации профиля засветки:

1. Предполагаем, что засветка распределена по квадратичному закону

Есть две контрольные точки этой зависимости:

(0,1.0) : точно известно, что при расстоянии 0 от края засветки значение GrIx = 1

(Rsb, GrIx min) – локальный минимум параболы, который будем находить на каждом снимке адаптивно:

А. фильтруем шумы от краевых эффектов снимков;

Б. Фильтруем другие непредсказуемые шумы (конструкции судна, другие помехи) – для этого возьмем распределение GrIx по расстоянию от края засветки, разобьем на бины

Будем брать количество бинов равное максимальному расстоянию r до края изображения

в каждом посчитаем статистику и отфильтруем значения за пределами 3σ

В. На каждом бине в оставшихся данных возьмем минимум, по ним построим зависимость от r

Г. Определим минимум у зависимости, примем, что это минимум функции распределения засветки от расстояния r до края солнечной засветки

Д. Имея две контрольные точки (одна из них - особая) получим коэффициенты квадратичной зависимости, используем ее для компенсации засветки

Поправки:

1. После этапа «В» вставляю сглаживание скользящим средним с экспоненциальным весом. Беру окно в 21 узел, показатель степени экспоненты -0.4
2. Начиная с пункта Г – поступаю по-другому. В процессе исследований стало ясно, что зависимость далека от квадратичной настолько же, насколько от линейной. Решено было аппроксимировать полиномом более высокой степени. Аппроксимацию провожу методом наименьших квадратов.

При этом одно из уравнений системы ЛУ, которая дает коэффициенты полинома, решено заменить на условие фиксированной точки на границе солнечной засветки: точка (0.0; 1.0)

Неплохие результаты дала аппроксимация полиномом 10 степени. Более низкие степени тоже показывают неплохие результаты, но при условии, что фиксированная точка сильно выбивается из ряда гладких данных, собранных на этапе «В» с первой поправкой, приходится взять степени повыше, чтобы компенсировать возникающие биения. Замечу, что нечетная степени вообще довольно редко показывала подходящий результат.

После таких операций вычитание из данных полинома, аппроксимированного по тренду минимума данных в зависимости от расстояния от края солнечной засветки, дает результат со следующими особенностями:

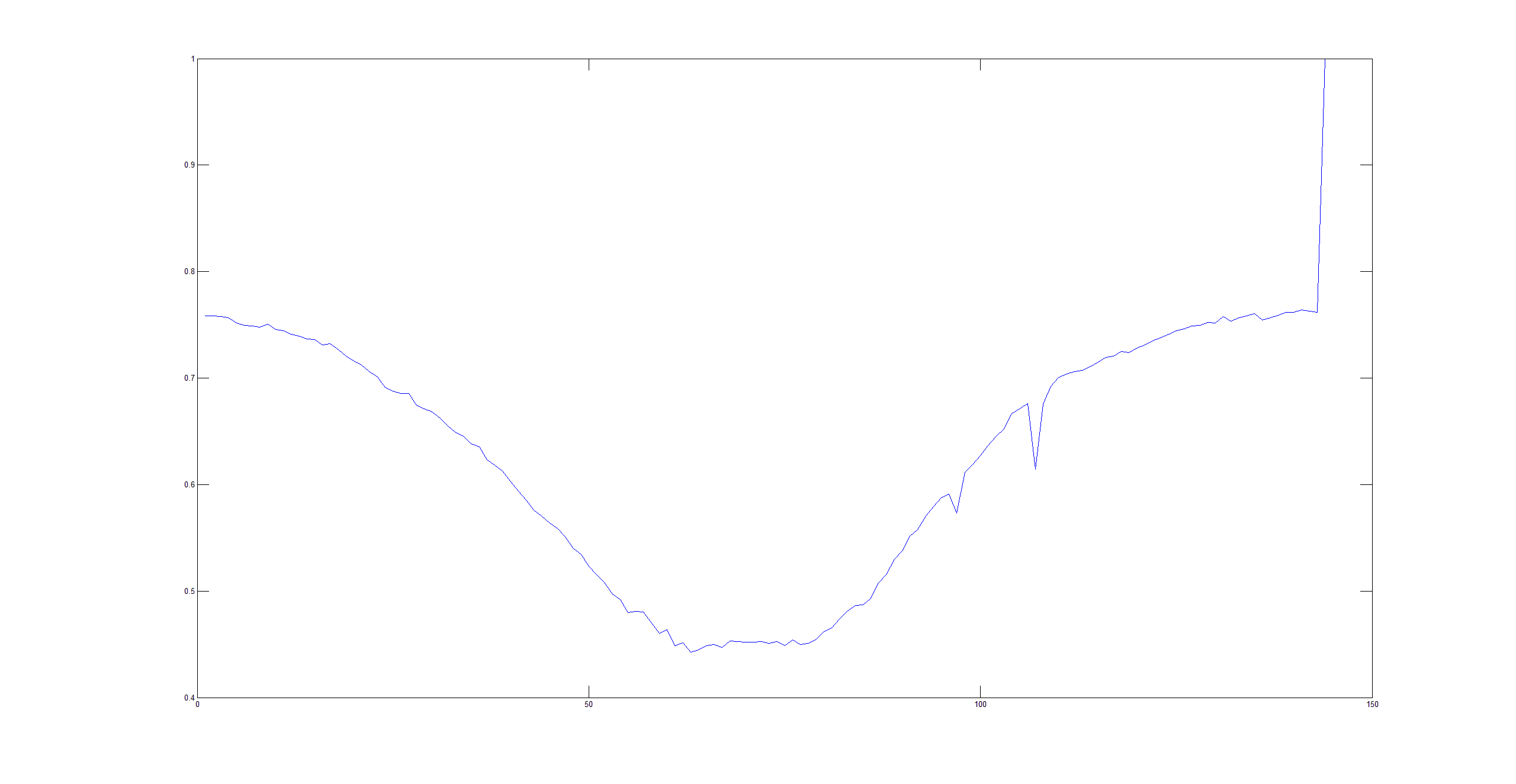
1. Солнце фильтруется очень хорошо
2. Хорошо выражены структуры облачности на средних расстояниях от края засветки
3. Сильно выпирают данные краевых завалов
4. Очевидно, что распределение оказывается все-таки анизотропным: в направлениях, в которых край изображения ближе к солнцу, убывание засветки проявляется меньше и минимум оказывается менее глубоким. Хотя паттерн поведения профиля кажется таким же. В результате на изображении выделяются направления, в которых фильтрация облачности довольно точна, но так же и направления, в которых фоновая засветка плохо отфильтровывается.

Получен результат распределения минимальных значений GrIx в пространстве (r,phi). Исследование показывает, что распределение действительно зависит от угла. Максимальное углубление фоновой засветки наблюдается в направлении, в котором край снимка дальше всего от солнца. При удалении от этого направления по углу отклонения локальный минимум смещается ближе к краю засветки и становится мельче.

Обнаружено, что в некотором приближении локальный минимум распределен по значению r линейно (если выбросить зависимость от угла). Проведем такие исследования.

В то же время есть предположение, что локальный минимум в направлении phi в зависимости от phi тоже имеет некоторую вполне определенную зависимость. Есть предположение, что можно ее аппроксимировать некоторой тригонометрической функцией. Проведем такие исследования.

Примерная зависимость минимумов от угла:



Похожа на sin^2(phi) при смещении нуля по углу в минимум зависимости.

Примерная зависимость минимумов от расстояния до центра солнца:



Очевидно, что это не линейная зависимость, но в определенном приближении можно счесть линейной.

Провести аналогичные исследования на других снимках.

Если картина выяснится похожая, - сделаем так:

При анализе снимка будем строить такие же зависимости:

1. Переводим данные в полярные координаты с центром в центре солнечного диска
2. В каждом узле ищем минимум с фильтрацией выбросов (за пределами 2\*sigma)
3. Разбиваем пространство углов на какое-то ощутимое для статистики количество узлов
4. По каждому значению угла находим локальный минимум зависимости данных, полученных в п.2. Локальные минимумы в этом исследовании будут иметь две координаты: расстояние и угол.
5. Аппроксимируем распределение по углу функцией a+b\*sin^2(phi-phi\_0), отсюда получим phi\_0, a, b
6. Аппроксимируем распределение по r линейной функцией по МНК.
7. В результате получаем зависимость: для каждого направления phi получаем значение локального минимума и по обратной зависимости – значение r, в котором этот минимум должен быть достигнут.
8. ПРЕДПОЛОЖУ, что зависимость значения GrIx фоновой засветки от r в каждом направлении одна и та же, только масштабируется как по r, так и по GrIx. Тогда получу зависимость для каждого угла phi следующего характера:

k = r1/Rs

GrIx1 = 1-k(1-GrIxs)

Судя по этой схеме результат должен получиться пересечением конуса с плоскостью сечения.

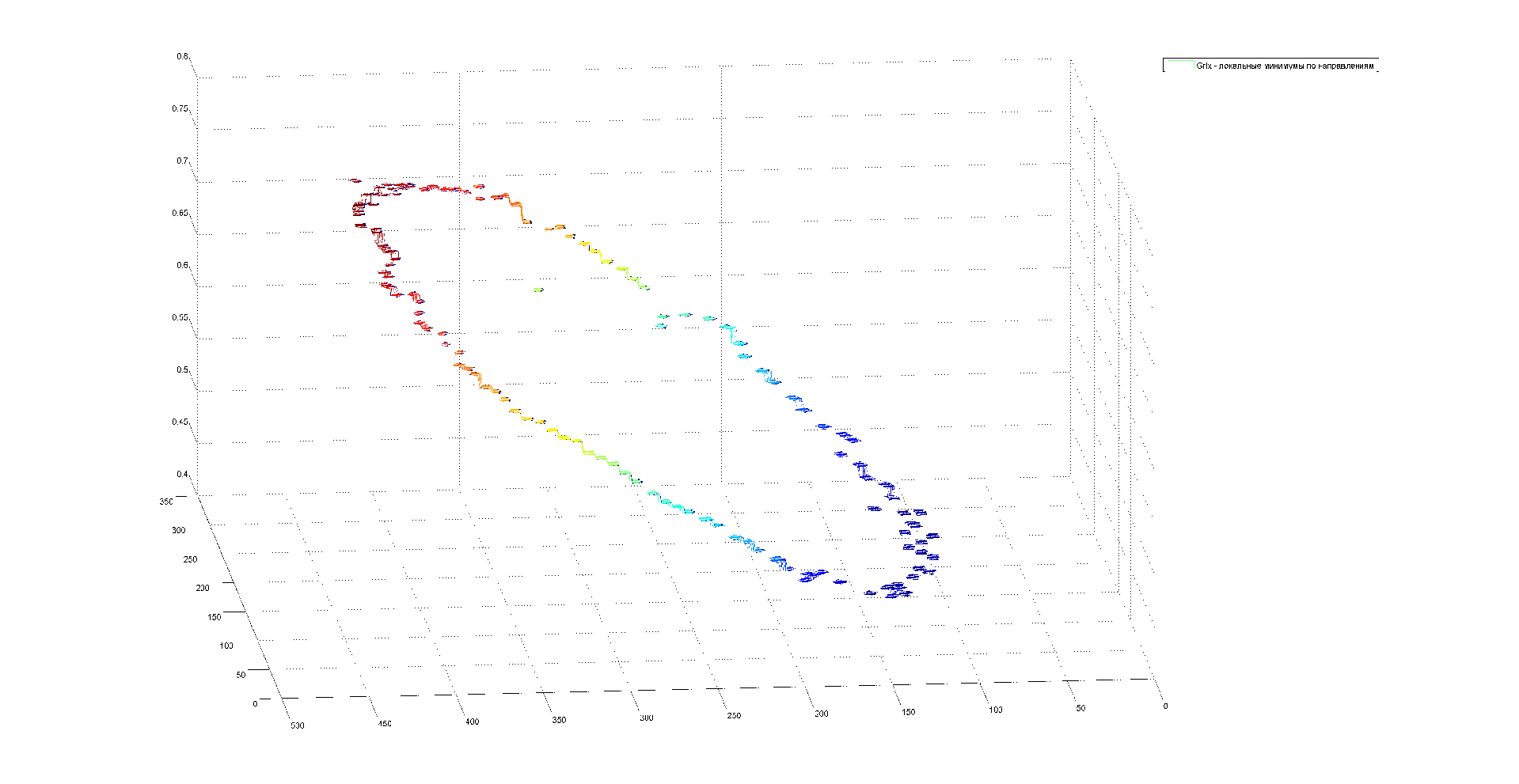
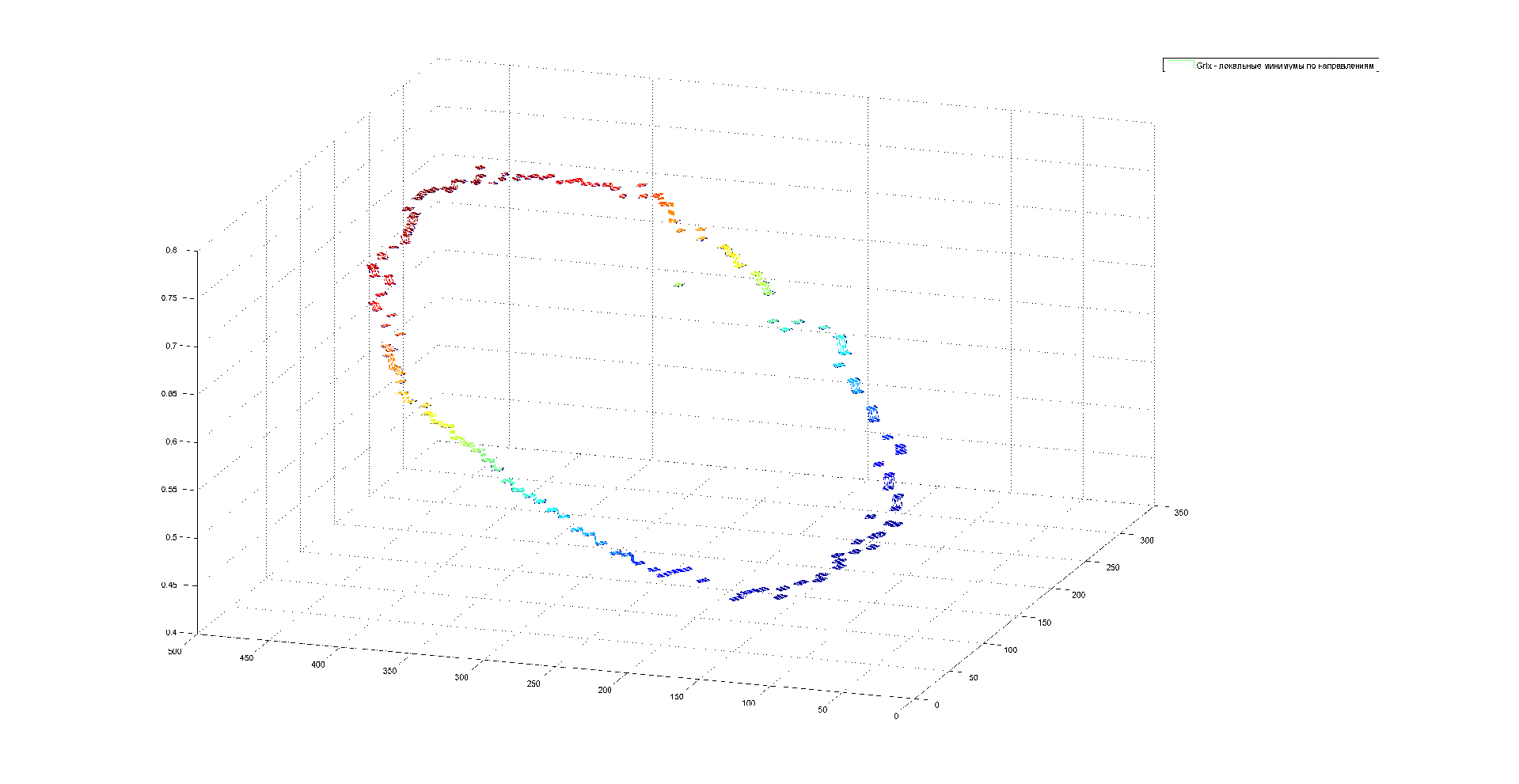
Конус в пространстве (x,y, GrIx) должен быть с центром в центре солнца и образующей наклоненной к горизонтали GrIx = const под углом theta, tg(theta) = GrIxs/Rs

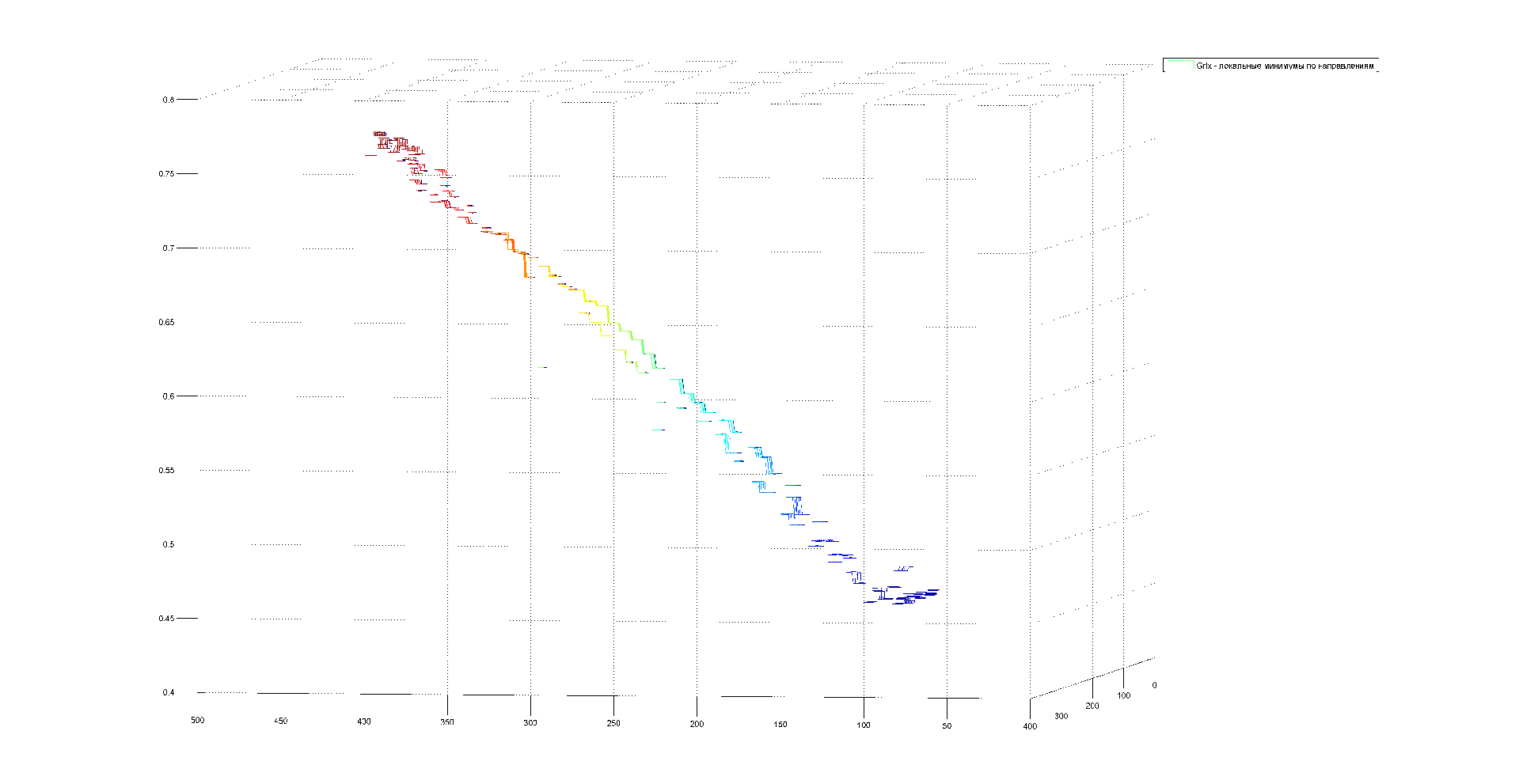
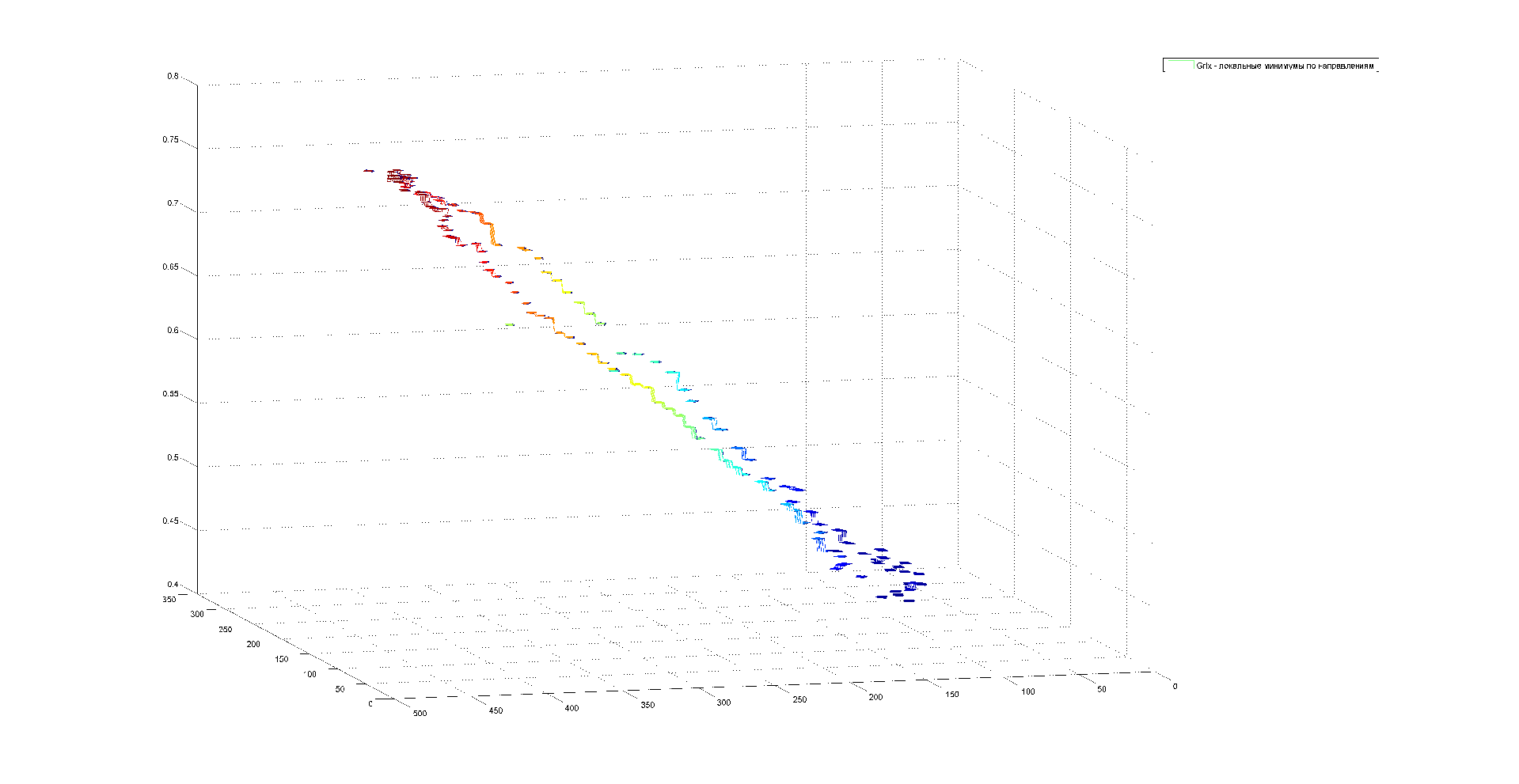
Секущая плоскость проходит через ту же точку положения глобального минимума (Rs), наклонена к горизонтали GrIx = const под углом gamma, tg(gamma) = (GrIxs – GrIxгgmm)/(Rs + rgmm), где точка с индексом gmm – локальный минимум в направлении противоположном направлению на глобальный минимум.

Не. Чета очень сложно.

Исследуем минимумы на предмет распределения в декартовом пространстве.

Первоначальное исследование показывает удобный результат (это одни и те же данные в разной перспективе)





Проверим, показывают ли похожий тренд другие случаи

Да, показывают. Значит, пробуем модифицировать подход учитывая этот факт:

Масштабируем зависимость по GrIx и r от края засветки. Для этого сначала оценим распределение минимумов: похоже, что все они лежат в одной плоскости, проходящей через точку глобального минимума, что должна лежать в направлении самого дальнего от солнца края видимой области снимка.

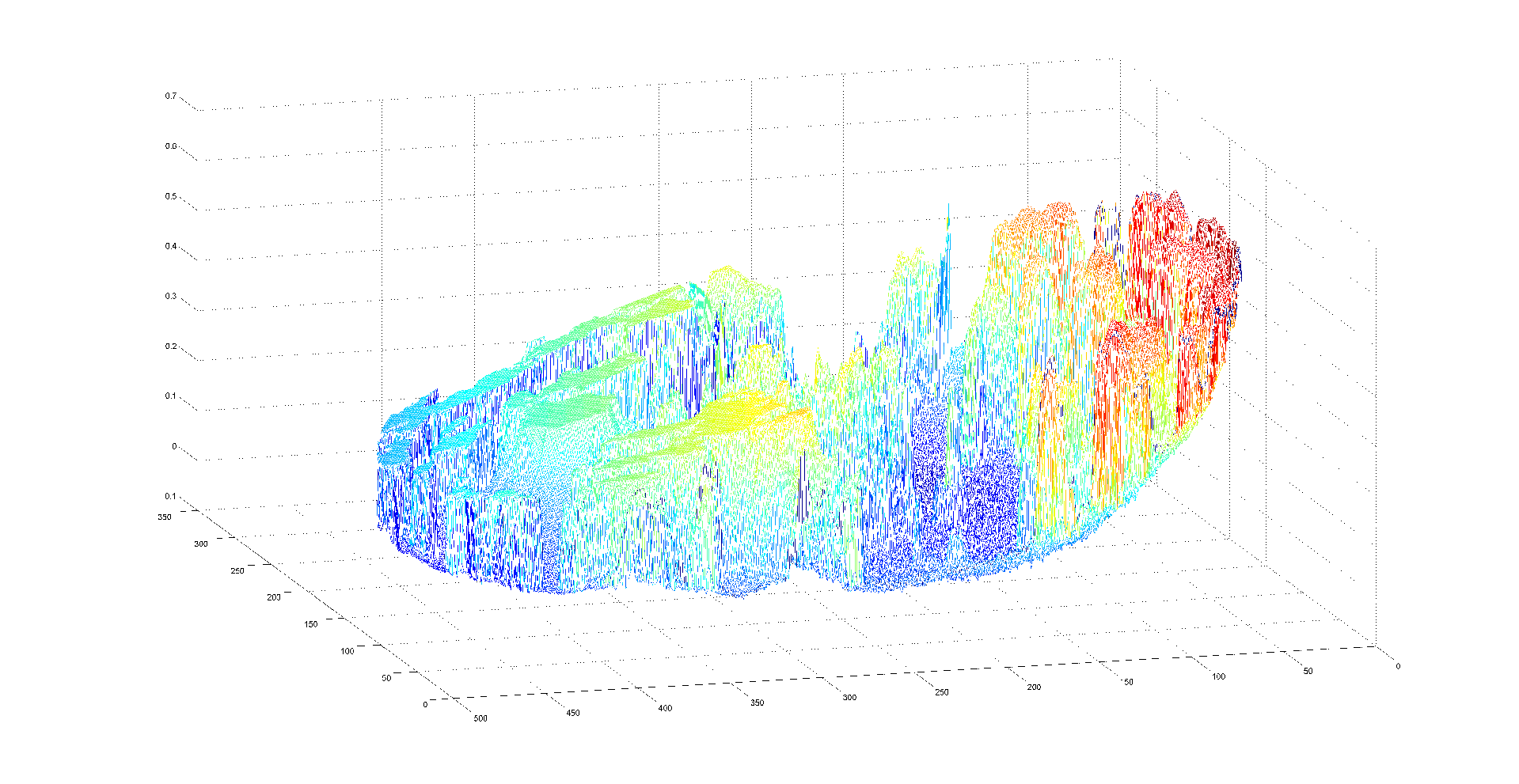
1. Определим направление этой прямой в координатах (x,y) – эта прямая проходит через точку глобального минимума и центр солнечной засветки
2. Пересчитаем координаты всех минимумов с учетом нуля в центре солнечной засветки – это будет координата Xs
3. Аппроксимируем зависимость значений минимумов от Xs прямой по МНК

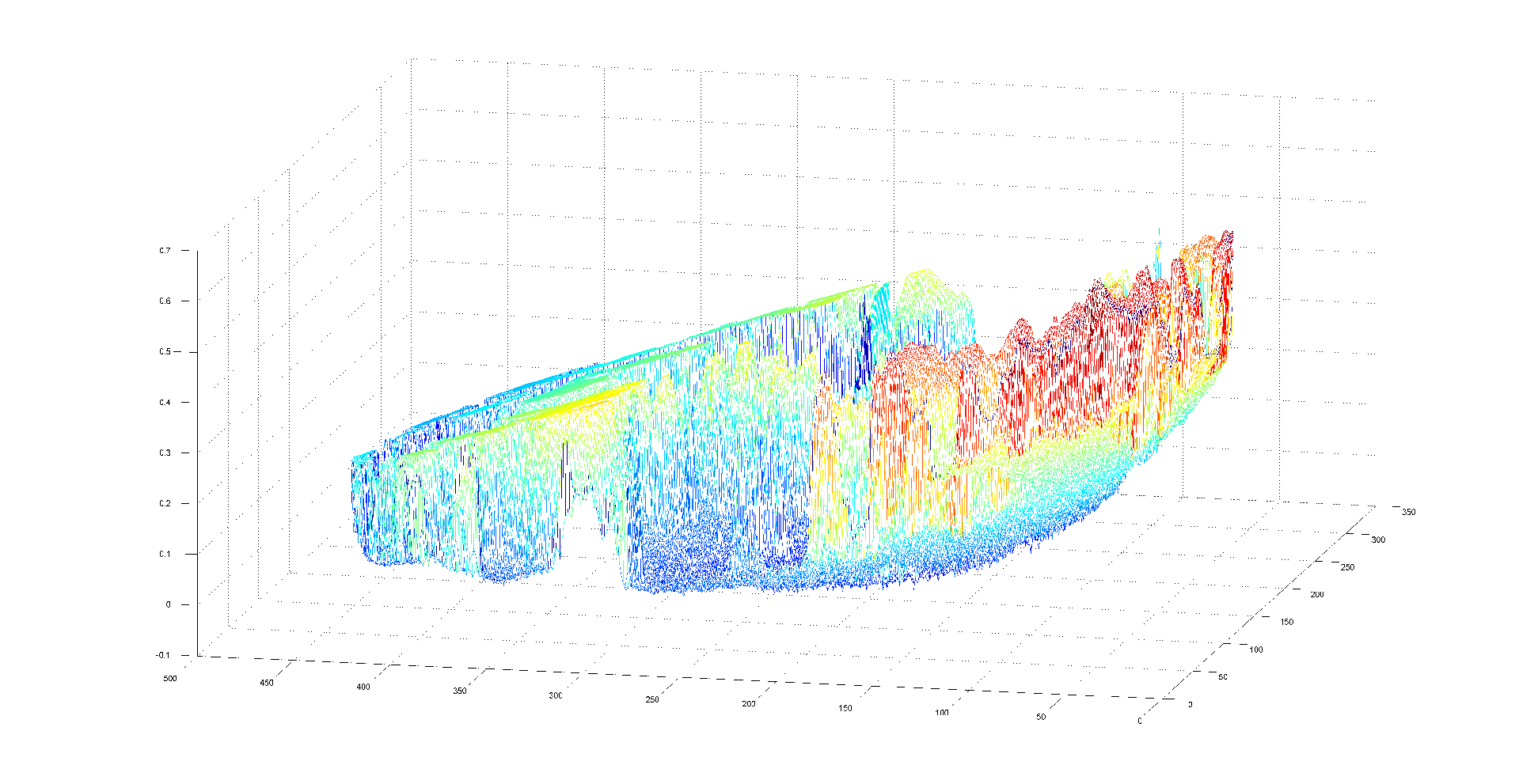
Получается, что всего лишь подтверждено, что локальные минимумы примерно укладываются в одну плоскость.

КСТАТИ!!! А не вычесть ли нам этот тренд в целом из всего поля???!!!!

Исследуем этот вопрос!

Исследовал. В результате сильно задирается край, дальний от солнца.





Попробуем, раз уж рассчитал распределение минимумов данныз в различных направлениях, поработать все-тки с ним. Известен тренд распределения локальных минимумов этих минимумов в направлении координаты Xs.

Пересчитаем углы от направления (центр солнца – центр изображения - глобальный минимум)

Нашли смещение по углу Phi1. Все углы в рассчетах надо будет модифицировать на эту величину.

Последовательность действий:

1. Имеем зависимость значения локального минимума от Xs
2. Оценим зависимость от phi при смещенном на phi1. Для этого нужны:
   1. Минимальное значение (оно есть – глобальный минимум)
   2. Максимальное или среднее значение. Достовернее, наверное, среднее. И это будет половина амплитуды
3. Для каждой точки:
   1. Считаем угол
   2. Смещаем на Phi1
   3. Вычисляем значение GrIx локального минимума для данного направления (см.п.2)
   4. По зависимости GrIx(r), а вернее обратной, найдем расстояние rs для локального минимума в этом направлении
   5. В предположении о линейном масштабировании зависимости фонового значения засветки от r в каждом направлении, используем полученное значение rs для получения коэффициента масштабирования и по GrIx, и по r

Вариант: не находить rs, вместо него для коэффициента масштабирования брать расстояние от центра солнца до края круга кадра (не до края изображения!)

Изменения.

Исследование показывает, что зависимость r(phi) хорошо аппроксимируется зависимостью:

rs(phi) = k\*(D cos(phi) + sqrt( R2 – D cos2(phi) ))

Выражение за исключением умножения на k есть решение задачи расстояния от точки внутри окружности, смещенной от центра на расстояние D (угол phi отмерен от направления на центр окружности) до каждой точки окружности. R - радиус окружности.

k = Rs / (D+R)

В применении к нашей задаче получаем функцию с четырьмя параметрами:

rs = d1 cos(ф-ф0) + sqrt(d32 – d22 cos2(ф-ф0) )

Параметр ф0 введен благодаря тому, что исследование показывает неточное совпадение фазы аппроксимационной функции (направление ф = 0 берется как напралвение на самую дальнюю от центра солнца точку окружности) и зависимости rs(ф), получаемой в данных.

Аппроксимируем rs(ф) методом градиентного спуска.

Первичный набор параметров d1, d2, d3, ф0 возьмем из соображений, что

1. ф0 – смещение направления через центр солнца и центр окружности, представляющей снимок
2. d1 = d2 = Rs
3. d3 = k \* R

2014-01-12

Реализована аппроксимация методом градиентного спуска с контролем точности

Переделан алгоритм перевода в полярные координаты с центром в центре солнца. Работает на два порядка быстрее, разрешение теперь не играет роли.

Получил картину распределения локальных минимумов в каждом направлении. Теперь явно вылезли минимумы, которые не являются экстремумами – это минимумы на границе изображения. В статистике зависимости они участвовать не должны – их надо отфильтровать.

2014-01-15

В первом приближении алгоритм работает

Результаты очень хорошие, эмпирически довольно высокая точность определения границ облачности

ПРОБЛЕМЫ к исправлению:

1. Недостаточно высокая точность определения положения центра и границ солнечного диска. Это ведет к понижению точности распределения локальных минимумов по углу с центром в центре солнечного диска
2. Недостаточная точность определения зависимости в краевых случаях – когда солнце довольно близко к краю снимка. Надо решить, что делать в таком случае с зависимостью.
3. Алгоритм аппроксимации в некоторых случаях может «разнести», хотя формально он даст верный результат. Пример – значения:

D1 = 53.117

D2 = 25.211

D3 = INF

Phi0 = 3.988

Очевидно, что необходимо ставить какие-то условия, проверять их на каждом шаге аппроксимации. Если они начинают нарушаться – отключать продвижение по переменной, нарушающей эти ограничения.

1. Аппроксимированная зависимость в сторону глобального минимума работает довольно хорошо ровно до тех пор, пока она приближает реальные данные. Есть область данных, на которой она не подтверждена экспериментально. На таких расстояниях от края солнечной засветки надо использовать более простые решения: там, где зависимость не подтверждена аппроксимируемым массивом данных, надо менять тактику. Вычитаемые данные следует интерполировать линейно с учетом крайнего аппроксимированного значения и с условием, что на границе снимка значение должно уходить в 1.0 – в абсолютную белесую пелену за счет аэрозолей и пыли.
   1. 2014-01-15. Поправка: этот вариант работает не очень хорошо. Лучше просто оставить до края значение равное крайнему аппроксимированному с уклоном аналогичным общему уклону локальных[ минимумов.

Исправлена проблема п.3. Причина была в том, что в некоторых случаях при вычислении значений градиента минимизируемой функции механизм автокалибровки парциальных приращений в пространстве параметров приводил к неограниченному росту (или убыванию в ноль) этих приращений. В результате можно было получить компоненты градиента равные нулю или бесконечности, что вызывало на следующем шаге продвижение по соответствующему параметру на шаг бесконечной величины. Само по себе это не вызывало исключения, поскольку автоматически в этом случае минимизируемая величина приходила в бесконечное значение, формально не отличаясь на соседних шагах, что давало условие окончания аппроксимации как найденный локальный минимум. Это конечно же ошибочное решение. Ошибка была исправлена введением масштабных значений для параметров и ограничением шагов по каждому из параметров относительно этих масштабных значений.

Самое главное теперь – уточнить алгоритм определения положения и размера солнечного диска.

Реализация алгоритма определения положения и размера солнечного диска.

Просто определение засветки и его центра масс и радиуса от площади не катит – слишком часто бывает так, что солнце высвечивает облака и результатом выходит некорректное определение необходимых параметров.

Вместо этого используется другой подход:

1. Определяются слитные объекты засветки
   1. Объекты фильтруются по площади с ограничением как минимального, так и максимального относительного значения
2. Считаются следующие метрики этих объектов:
   1. Градиент GrIx (величина без направления) на границе. Для каждой точки границы берется усредненное значение в области, ограниченной расстоянием 10px. Внутренние для контура объекта точки не учитываются.
   2. По среднему значению градиента на границе выбирается наиболее подходящий (с минимальным значением)
   3. Среди точек границы полученного объекта отбираются такие, которые подходят в качестве границы солнечного диска. Условие отбора – значение градиента меньше среднего по границе этого объекта. Обосновывается это тем, что градиент на границе солнца обычно гораздо меньше, чем на границе облака
   4. По отобранным точкам аппроксимируется положение и размер окружности, которая в результате будет представлять солнечный диск. Аппроксимация проводится модифицированным методом градиентного спуска..

Для этого подхода был реализован модифицированный метод градиентного спуска, который приближает к нулю некоторую функцию в пространстве параметров. Особенностью модифицированного метода является необходимость задать напрямую функцию, которая будет представлять минимизируемое в пространстве параметров значение (в отличие от «базового» варианта метода, который рассчитан на то, что н вход передается функция, которая должна приблизить по сумме квадратов отклонений некоторый набор экспериментальных данных. В «базовом» варианте рассчет отклонения аналитической функции от экспериментального ряда заложен в метод).

Результат положительный: положение и размеры солнца определяются с бОльшей точностью, чем раньше, что дает более высокую точность определения всех остальных зависимостей в алгоритме.

2014-03-18

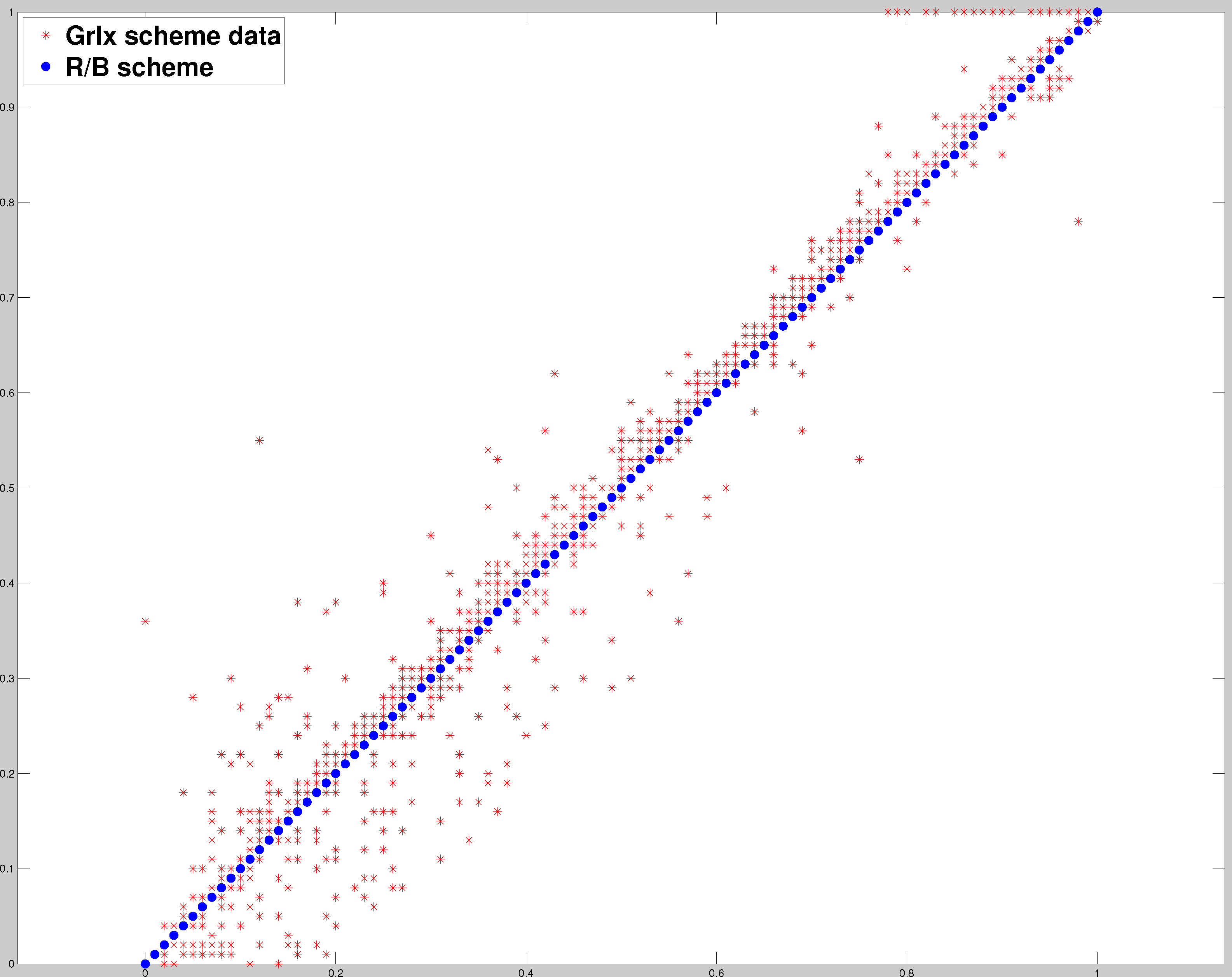
Запущена обработка большого количества снимков (1347) для набора статистики. Рассчитываются процент облачного покрытия по трем алгоритмам: японскому, немецкому и нашему.

В процессе выявлены и решены следующие особенности

1. Проблема с определением геометрии полезного изображения в кадре.
   1. Причины. Алгоритм был основан на определении трех краевых точек, по которым можно строить окружность. Краевые точки искались на прямых, соединяющих углы изображения. Причина некорректной работы алгоритма – ложное срабатывание нахождения второй точки при движении вдоль одной из прямых. Это происходило, например, когда маска изображения могла быть с дефектами по краю, что давало «выключенные» точки в непосредственной близости от определенной точки «включения».
   2. Решение: Изменен подход. После получения маски изображения ищется ее контур. Точки контура фильтруются по условию нахождения не ближе 5px от края изображения. Аппроксимацией находится наиболее подходящее положение и размер окружности, подходящей под эти точки. Используется аппроксимация методом градиентного спуска (модифицированная, с прямым указанием минимизируемой функции – суммы квадратов отклонений аналитически задаваемой окружности от полученного набора точек границы маски)
2. Проблема определения положения и размеров солнца при его очевидном наличии и простом расположении
   1. Причины. Пока неясно.
   2. Решение. Пока неясно.
3. Проблемы крайних случаев вроде расположения солнца слишком близко к горизонту или за пределами полезной области снимка
   1. Причины. При расположении солнца вблизи края снимка, но за его пределами (особенно если кадр обрезанный) может случиться так, что диск солнца определяется, его центр лежит за пределами снимка. И при этом может получаться так, что все остальные элементы алгоритма тоже работают: отрабатывает нахождение минимальных значений по каждому направлению вокруг солнца, отрабатывает аппроксимация зависимости расположения локального минимума в каждом направлении в зависимости от угла. Но при этом вполне очевидно, что вследствие пограничности случая зависимость расстояния до минимума от угла может давать нефизичные (отрицательные) значения на некоторых значениях угла. Результатом чего будет являться сбой алгоритма на следующих стадиях.
   2. Решение. В настоящий момент используется существующее решение. На значениях угла, на которых аппроксимирующая функция дает нефизичное значение расстояние до локального минимума, вместо этого значения используется ноль. Это влечет за собой отсутствие на этих направлениях части вычитаемого тренда, получаемого масштабированием из общего, определенного в полярной системе координат по всему изображению в целом.

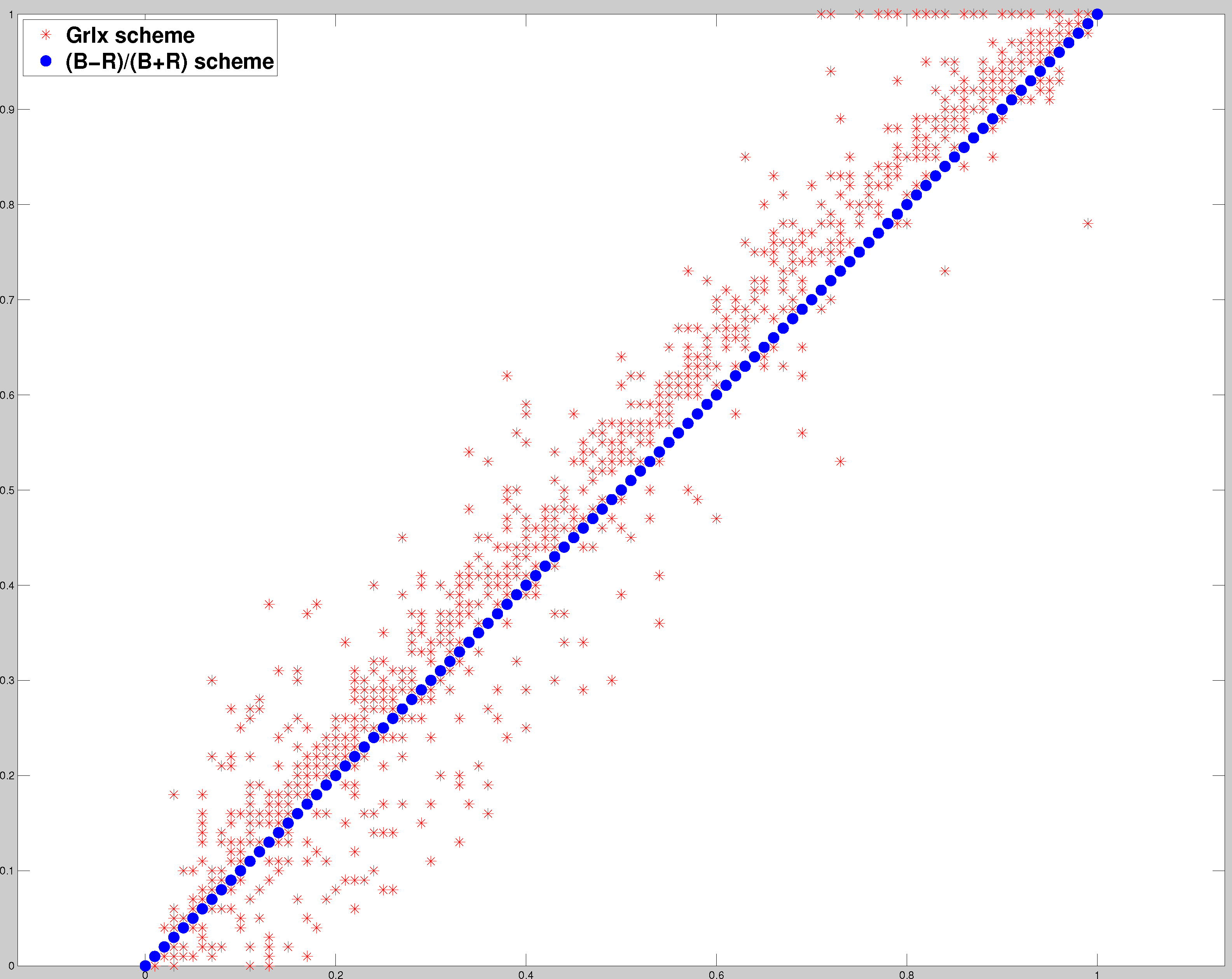
Результат прогона:

В среднем алгоритм, использующий индекс GrIx дает более высокую облачность, чем два других:



Здесь из 1347 снимков схема grIx дает большее облачное покрытие в 864 (71,5%) случаях и меньшее – в 483(29,5%) случаях.

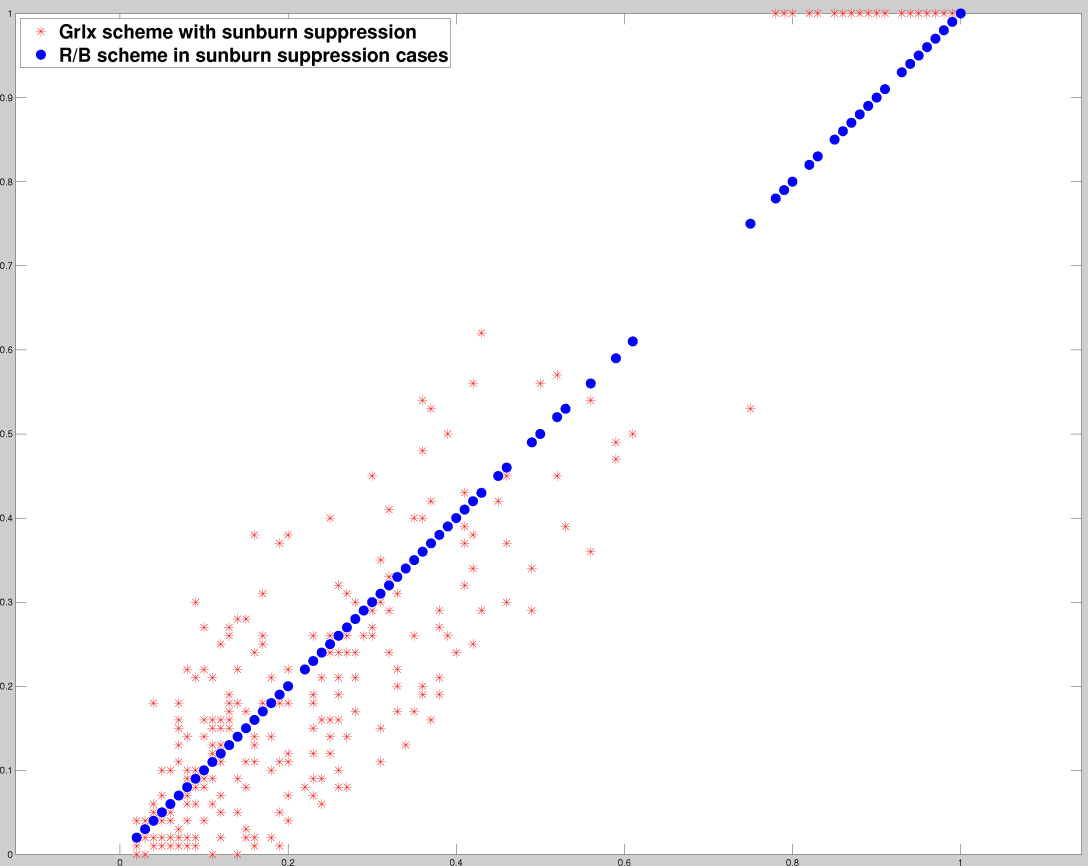
В сравнении с японской схемой ситуация еще ярче: схема grix дает большее облачное покрытие в 1056 (78,4%) случаях из 1347. И лишь в 291 (21,6%) случаях – такое же или меньшее:



Интересно было посмотреть, как влияет на такую статистику факт подавления солнечной засветки.

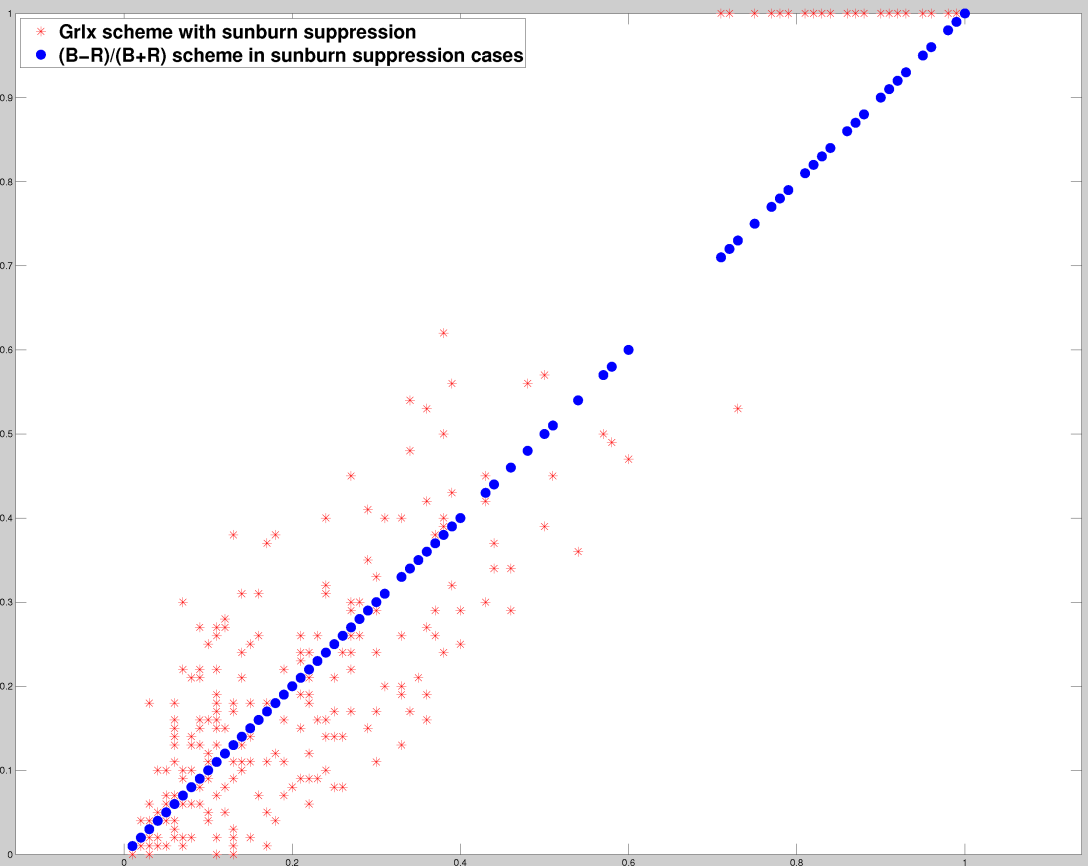
Получилось следующее:

Всего случаев с подавлением солнечной засветки: 287. Из них схема GrIx дает большее (чем немецкая R/B) облачное покрытие в 123 (42,8%) случаях и меньшее – в 164 (57,2%) случаях.



При сравнении с японской схемой ((B-R)/(B+R)) картина следующая:

Из 287 случаев схема GrIx дает бОльшее облачное покрытие в 143 (49,8%) случаях и мЕньшее – в 144 (50,2%) случаях.



Поскольку коэффициент корреляции между значениями по японской и немецкой схеме почти единица (0,997), можно для сравнения в будущем использовать одну из этих схем или среднее между ними. Использую среднее.

+++++++++++++++++++++++++++++++++++++++++++++++++++

Пока не реализовано – потребность отпала методически

+++++++++++++++++++++++++++++++++++++++++++++++++++

Идея по улучшению алгоритма аппроксимации с использованием оптимизации методом градиентного спуска. При условии, что нужно использовать ограничения на пространство параметров, можно сделать так:

1. Напомню, минимизируем некоторую функцию f(**p**) – функцию суммы квадратов отклонений исходной аппроксимирующей функции F(**p**, x) от набора экспериментальных данных.
2. Получить «запрещающую» функцию g(**p**), условие на которую будет ограничивать пространство параметров:

g(**p**) > 0

1. Проверять ситуацию нахождения следующей точки pi+1 в разрешенной области. Если эта точка находится в запрещенной – не шагаем туда. Вместо этого:
2. Ясно, что сейчас мы находимся в точке **p**i, близкой к гиперповерхности g(**p**) = 0
3. Хорошо бы найти касательную гиперплоскость, спроецировав на которую вектор антиградиента **ag** = –grad( f(**p**) ) получим продвижение из точки pi в нужную сторону, но параллельно поверхности, ограничивающей запрещенную область.
4. ПРЕДПОЛОЖЕНИЕ: касательные к гиперповерхностям (g(**p**) = 0 +- d) тем точнее параллельны, чем меньше d. Это значит, что вектор градиента g(**p**) нормален к касательной к гиперповерхности (g(**p**) = 0). Ищем единичный вектор **n** = **grad** (g(p)) / |**n**| в точке pi (кстати он направлен в область дозволенных значений **p** – в область положительных значений g{**p**}, ибо это градиент)
5. Дальше имея вектор антиградиента **ag** и нормаль **n** легко находим новый вектор перемещения:

**ag**’ = **ag + n \*** | (**ag,n**) |

следующий шаг должен быть вдоль этого вектора

+++++++++++++++++++++++++++++++++++++++++++++++++++

Пока не реализовано – потребность отпала методически

+++++++++++++++++++++++++++++++++++++++++++++++++++

Идея по улучшению точности определения процента облачного покрытия в условиях, когда не выявлена возможность подавления солнечной засветки.

Предпосылки: пока что не определяется, каков тип облачности и насколько хорош видно небо. При обложной тонкой облачности можно было бы говорить о пониженном балле, а при яркой кучевой облачности – считать по стандартной схеме.

Для того, чтобы определить тип действий, можно попробовать следующее:

1. Выделить зоны гарантированной облачности (grix > 0.9) и гарантированного неба (grix < 0.65)
2. Оценить остальную площадь. Сравнить с общей площадью облачности, с общей площадью снимка, с общей площадью неба. Попробовать по этим данным оценить, насколько обширны зоны промежуточных значений, из этого сделать вывод о типе облачности. Чем больше промежуточных значений, - тем больше вероятность, что это не кучевая резкая облачность, а тонкая обложная / дымка.
3. В случае дымки применять повышенное граничное значение. Оценить, какое.

При обработке большого количества снимков обнаружена неприятная особенность: реализованный алгоритм определения местоположения солнца недостаточно точен. Хочется его уточнить или заменить на более точный.

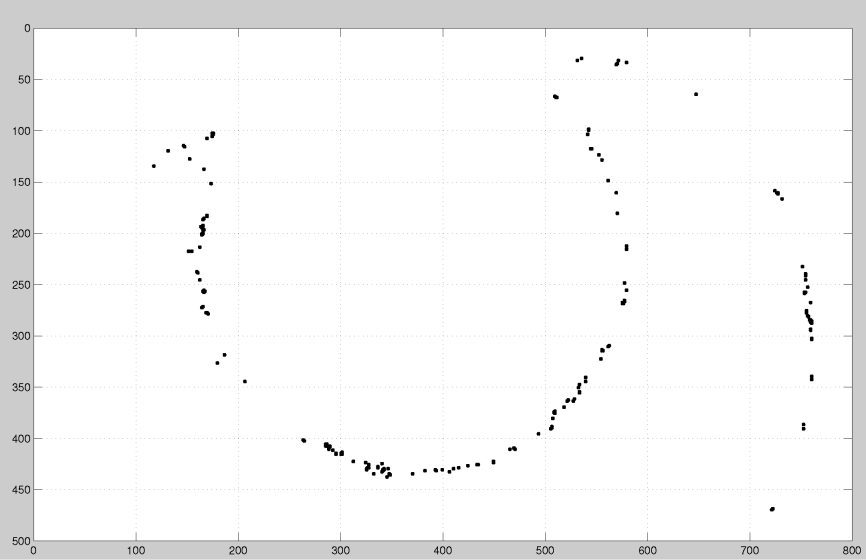
Исследую следующий вариант:

1. Исходные данные по grIx отфильтровать по принципу «оставить только гарантированное небо» - значения ниже 0.7.
2. Отфильтровать краевые шумы и шумы выбросов по цветности – оставить значения выше 0.5
3. Посчитать градиент поля (абсолютное значение без направления). Отфильтровать данные по значению градиентов – оставить значения поля grix только в узлах с достаточно высоким значением градиента. Так получим «склоны» образований на поле grix.
4. На полученном поле надо попробовать выделить изолинии (прерывистые, но они будут). Расположение точек изолинии в условиях центральной (солнечной) засветки будет подчиняться некоторой зависимости, похожей на окружность или на деформированый эллипс. Если предположить наличие в точке (x0, y0) некоторого искомого центра, от которого считается расстояние r до точек изолинии, они могут лежать на некоторой кривой, описываемой следующим соотношением:
5. Как видно, это соотношение разбивается на два – для x и y точек изолинии:

есть совокупность точек контура, которые следует аппроксимировать зависимостью, подчиняющейся этим соотношениям. Замечу, что пространство параметров здесь –

1. Пока нерешенная задача: отсечь точки изолинии, не относящиеся к нужной ее части. Очевидно, что по краю изображения могут быть данные с тем же значением grix.

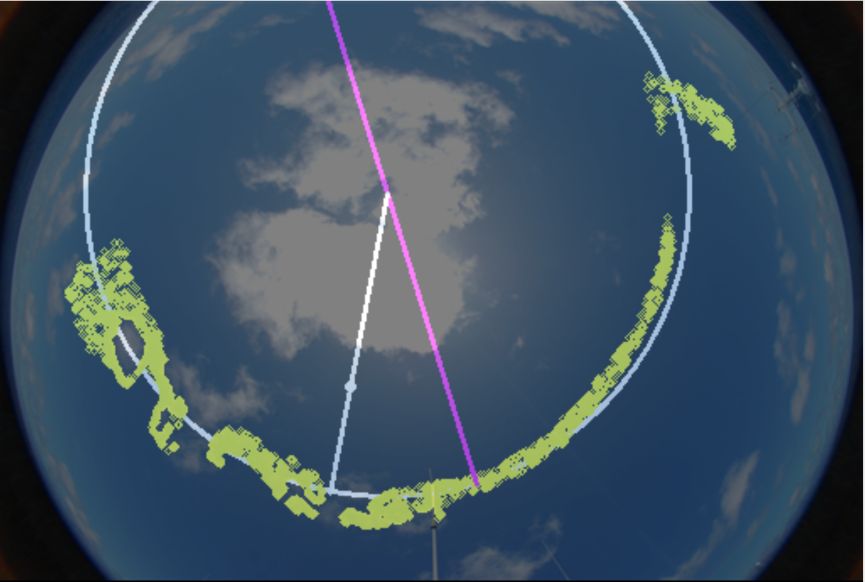
Кстати, а может попробовать приблизить просто окружностью???

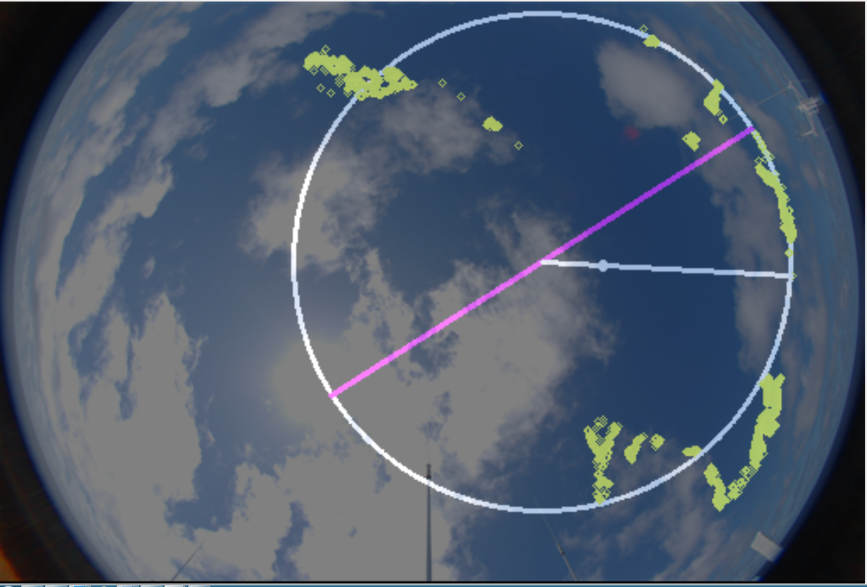


Окружностью – плохая идея. Зато вроде хорошая идея получается сделать следующим образом:

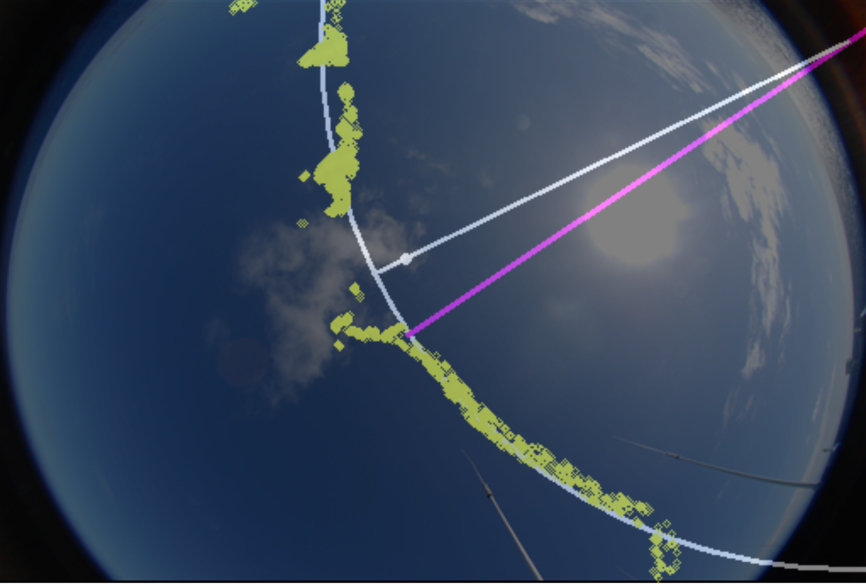
1. Обнаружить эти изолинии на фильтрованных данных
2. Аппроксимировать окружностью
3. Посчитать статистику квадратов отклонений от окружности по значению расстояния до центра окружности
4. Отфильтровать слишком далекие точки
5. Аппроксимировать профильтрованные данные окружностью еще раз
6. Пересчитать координаты точек в полярную СК с центром в центре окружности и нулевым направлением, например, направо.
7. Посчитать статистику по углу точек, отфильтровать статистически недостоверные выбросы
8. Взять дугу аппроксимированной окружности от минимального до максимального угла отфильтрованных данных
9. Провести прямую – биссектрису полученной дуги. Условия расположения солнца: лежит на этой прямой, внутри аппроксимированной окружности.

Получается нечто вот такое (красным выделена прямая, на которой ожидается расположение солнца):

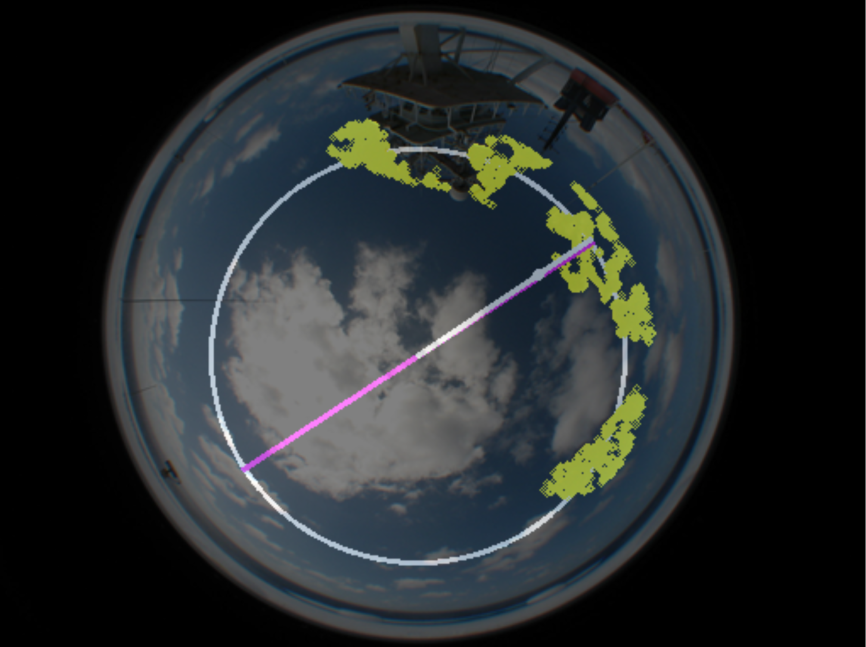








Более того, даже вот в таких вариантах можно оценить направление, на котором можно судить о расположении солнечного диска.



2014-04-17

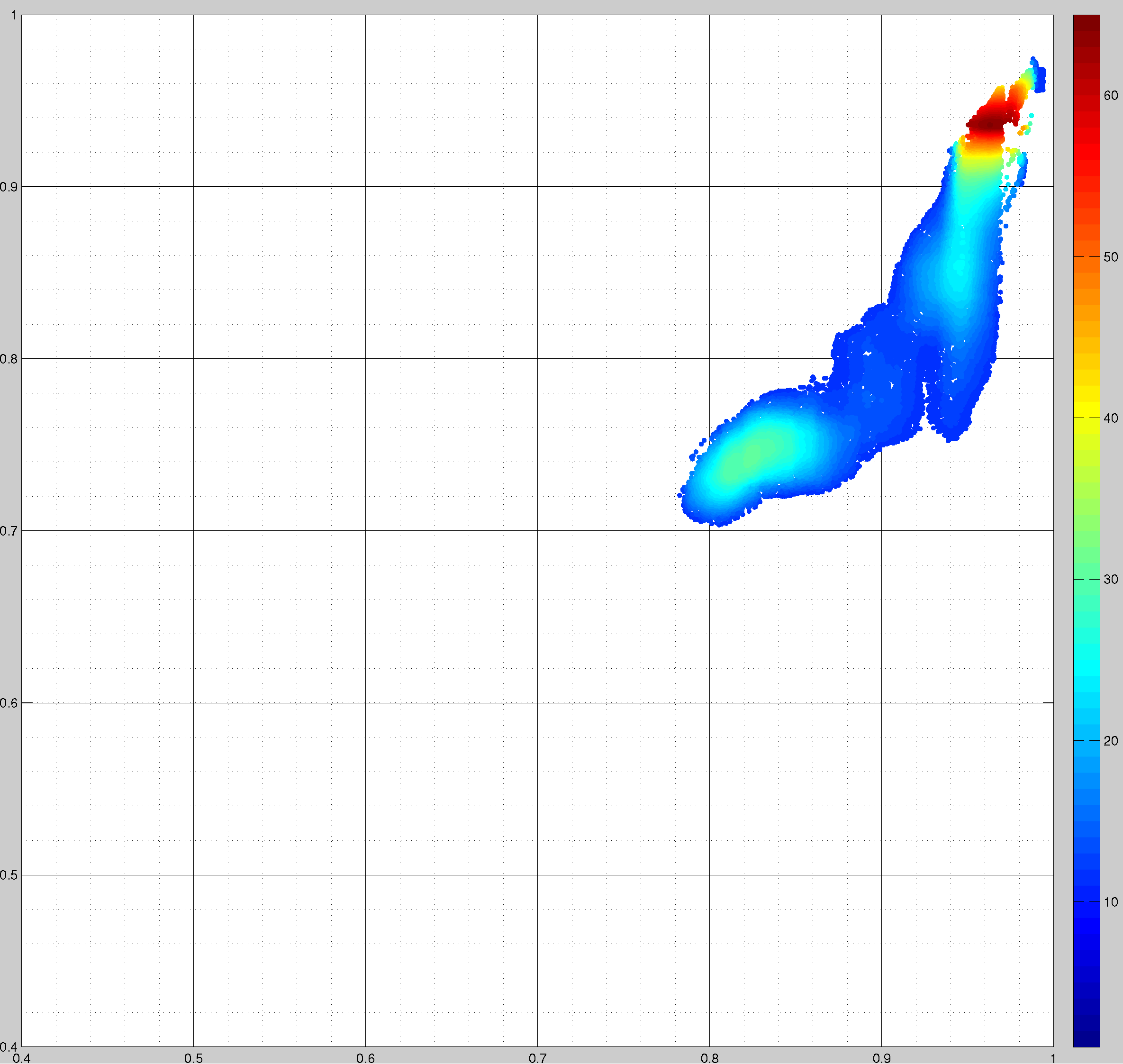
Алгоритм обнаружения солнца значительно модифицирован.

Оказалось, что предложенная схема, определяющая положение солнца по биссектрисе дуги, построенной на доступных точках изолинии (в пределах некоторого небольшого интервала значений GrIx), работает намного лучше и точнее (хотя и медленнее), чем схема, опирающаяся на градиенты поля GrIx вдоль границ областей засветки. Более того, в некоторых случаях получается так, что само солнце скрыто облаками, но по предложенной схеме его положение определяется достаточно точно. В случаях, когда влияние засветки в таком состоянии диска солнца велико, можно использовать алгоритм подавления эффекта засветки даже при условии, что самого диска солнца на снимке не видно.

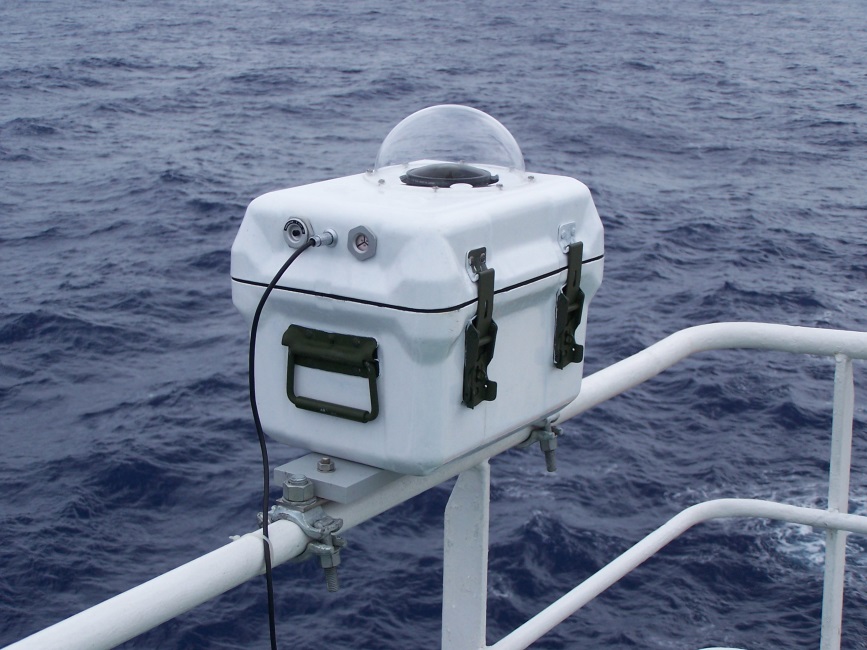
В связи со всем этим алгоритм определения положения солнца на снимке заменен на схему, использующую биссектрисы дуг изолиний поля GrIx.

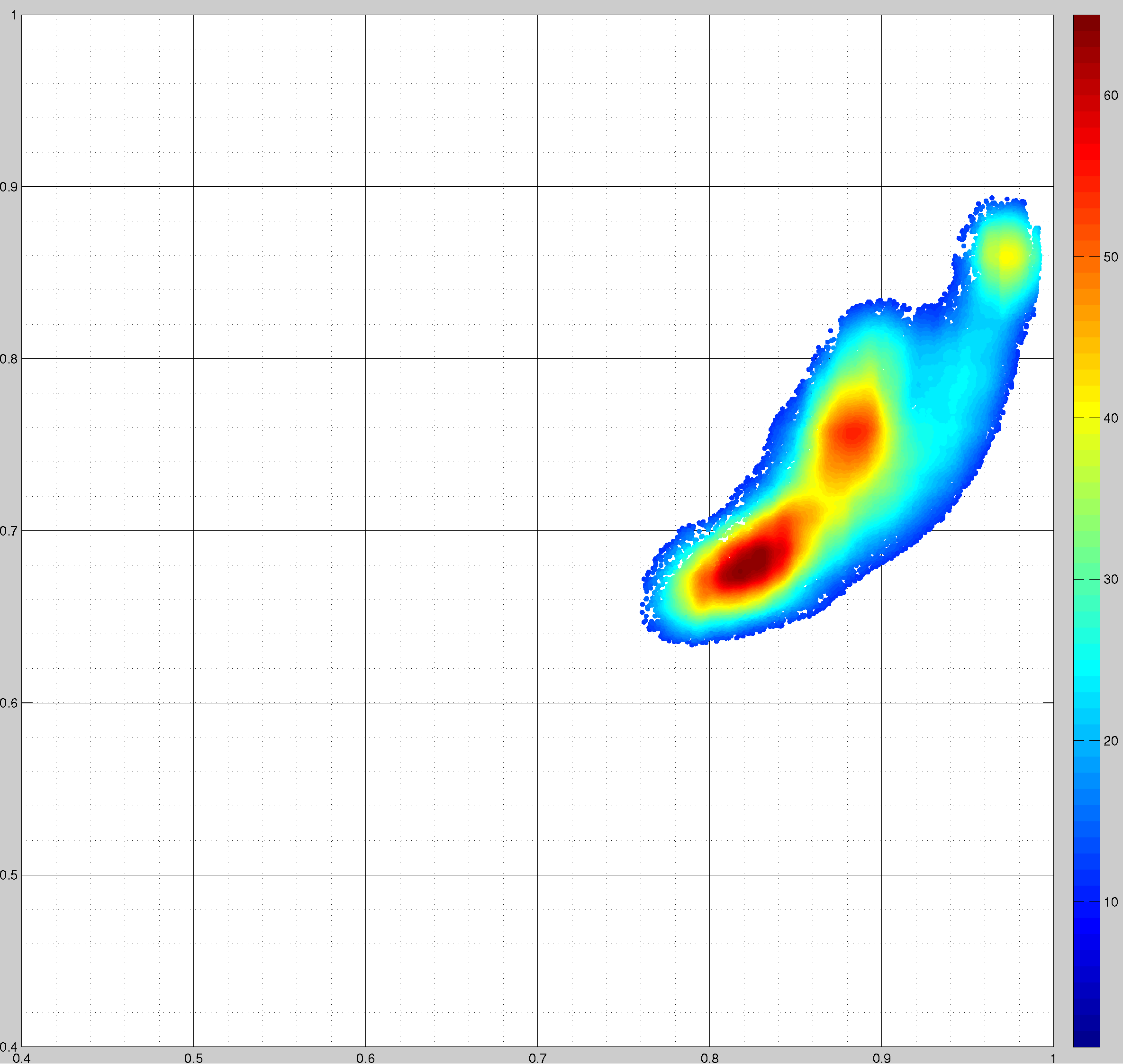
Следующим важным фактом стало то, что разное оборудование дает разные снимки, если оценивать их в общей массе. В результате этого граничное значение GrIx, подходящее для классификации точек «небо/облако» на снимках с одного оборудования, далеко не всегда подходит для снимков с другого оборудования. Влияние оказывают очень многие факторы: от динамического диапазона камеры до наличия фильтров или защитных стекол на камерах (последние, например, «замыливают» снимки, что смещает общую функцию распределения GrIx снимка в сторону 1.0). В результате можно прийти к выводу, что схема должна быть нормирована на оборудование.

Для исследования характерных особенностей съемки каждым набором приборов была проведена следующая работа: для каждого снимка в серии снимков каждой установки были посчитаны для поля GrIx медиана (как наиболее яркая характеристика распределения в целом) и 5-й перцентиль (как значение, которое предположительно характеризует нижний значимый порог значений распределения). Были построены скаттер-плоты 5-го перцентиля от медианы, которые дают для разного оборудования следующие картины:

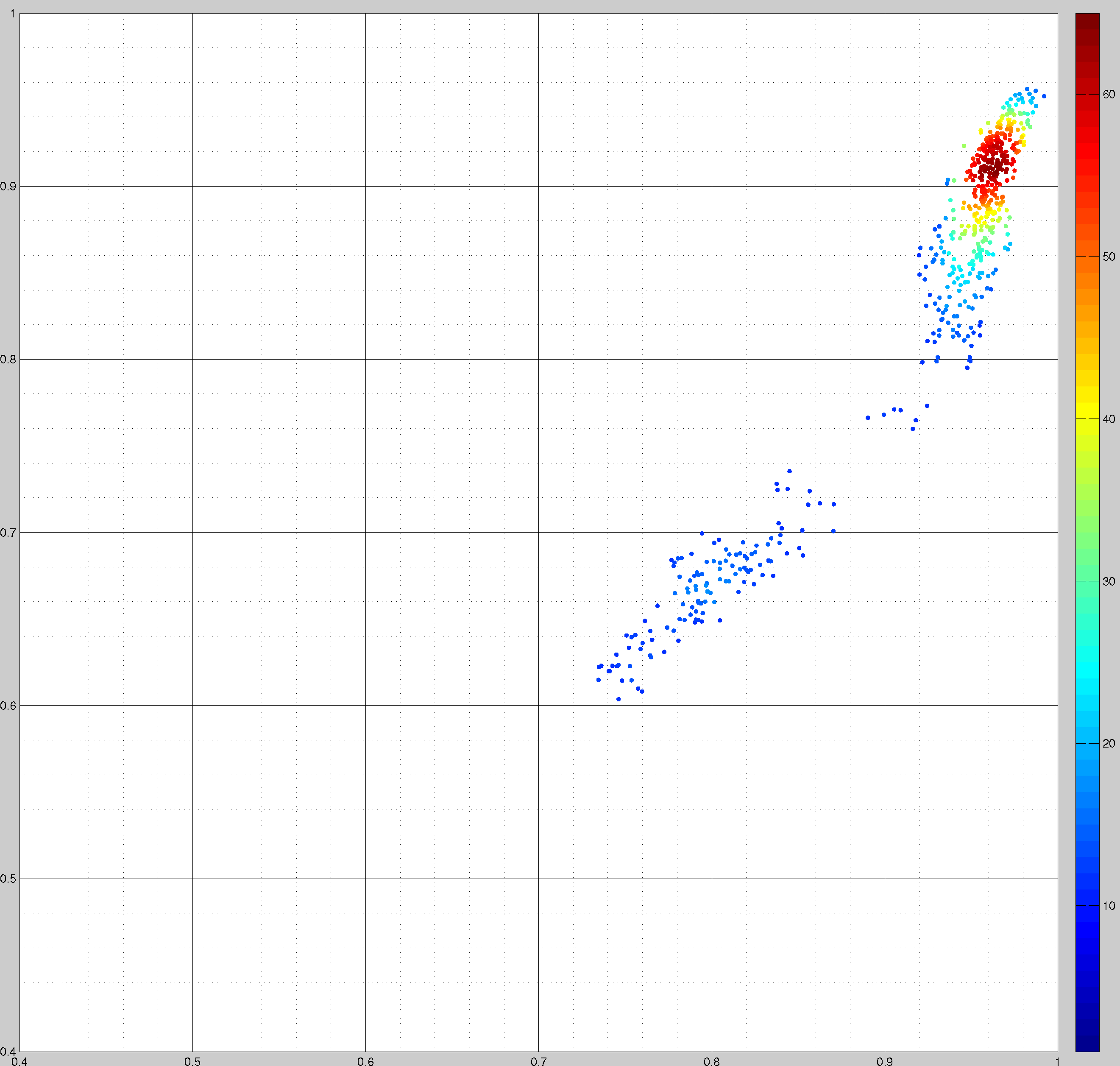


NIKON E8800 с защитным колпаком:





Та же установка, но с камерой Canon A640

В случае со съемкой с рук в рейсах на Иоффе использовался Nikon D80:

В этих точечных диаграммах отсечены шумы со значениями плотности ниже 10% от максимальной.

Диаграммы показывают общую для всех установок закономерность: есть области высокой плотности точек. В некоторых случаях заметны только две таких области – в зоне высокой облачности (близкие к 1.0 значения медианы и 5го перцентиля) и в зоне средней облачности (на этих диаграммах – зоны пониженного расположения медианы и 5го перцентиля). При достаточном большой выборке (для CanonA640 это 56808 снимков, для Nikon E8800 – 38124 снимка) могут быть заметны дополнительные зоны скопления точек.

Характеристика оборудования может быть получена из статистики описанного рода.

Минимально, что попробую сделать:

1. отсечь шумы (области с плотностью точек, например, меньше 10% максимальной)
2. оценить минимальные значения для оставшихся точек по медиане и 5му перцентилю
3. исходя из соотношения этих значений попробовать оценить смещение границы деления небо-облако
4. Кроме того, можно выделить зону верхнего скопления точек, которая позволит отнести снимок к обложной облачности и сразу предоставить решение о применимости алгоритма подавления засветки (не применять)
5. После набора статистики можно ориентироваться на средний тренд (посчитать для каждого распределения по МНК, например, аппроксимацию полиномом степени от 3 до 7) и по расположению конкретного случая относительно этого тренда и отклонения от него, принимать решение о характере данных на снимке. Примеры:
   1. Если снимок можно отнести к зоне близких к 1.0 медианы и 5го перцентиля, можно не подавлять засветку и сразу считать по схеме с простым делением точек по граничному значению GrIx
   2. Если точка лежит недалеко от аппроксимационной кривой, надо действовать длинным путем: оценивать положение солнца, подавлять засветку и проч
   3. Если точка лежит между биссектрисой диаграммы (выше просто не может – 5й перцентиль не может быть больше медианы) – значит, гистограмма распределения GrIx прижата в сторону медианы, а это означает один выраженный пик в районе медианы. Это значит, что в целом насыщенные цвета на снимке присутствуют больше, чем в среднем, что должно сигнализировать о некотором повышенном открытии области чистого неба. Здесь точно следует использовать подавление солнечной засветки
   4. Если точка лежит выраженно правее аппроксимационной кривой, - значит, по сравнению со снимками с таким же нижним порогом значений GrIx медиана смещена в сторону 1.0. Это означает, что распределение в целом обычное, но плотность смещена в сторону более серых значений. Это может означать загрязнения на защитном экране, дымку или любые другие факторы, «замыливающие» изображение. Кроме того, это может быть признак высокой прозрачной облачности, которую так сложно ловить и обрабатывать. В таких случаях обычно солнце есть в кадре, неплохо определяется, но плохо фильтруется. Кроме того, в таких случаях чаще всего солнце покрыто полупрозрачными облаками и область его расположения скорее следует считать областью с облаком, а не точным отсутствием облачности, как это происходит сейчас.
   5. Если точка находится в целом недалеко от аппроксимационной кривой, и смещена в сторону повышенных значений медианы, - значит, в целом снимок подчиняется общей закономерности, но при этом у него повышенная «серость». Скорее всего будет детектирован процент облачного покрытия выше 50%
   6. Если точка находится в целом недалеко от аппроксимационной кривой, и смещена в сторону пониженных значений медианы, - значит, на снимке велико содержание точек с насыщенной цветностью. Это самая удобная ситуация: есть и облака, и небо, и (скорее всего) солнце, которое легко найдется
   7. Есть еще интересная зона диаграммы, в которой встречаются значения, хотя их и не так много. Это зона с пониженным 5ым перцентилем и пониженной медианой. Точки, расположенные в этой зоне, сигнализируют о снимках с низким (достаточно насыщенным по цвету) порогом распределения значений GrIx на снимке, и в целом смещенной в сторону более насыщенной цветности гистограмме этого распределения. Такие факторы могут свидетельствовать о том, что «серость» снимка недостаточна. Это бывает, например, в вечерние часы, когда облака и прочие структуры подсвечиваются и окрашиваются. Это также бывает при сбое настроек баланса белого – тогда насыщенность цвета, соответствующего абсолютно белому (или абсолютно серому) на снимке выходит ненулевая. Эти случаи следует выбраковывать или исследовать отдельно.
6. Есть и другой способ классификации точек относительно набранной статистики. Сейчас пытаюсь его реализовать:
   1. Провести кластеризацию всех точек статистики
   2. По полученным кластерам сформировать классы и относить вновь появляющиеся снимки к этим классам. В зависимости от отнесения точки к тому или иному классу, следует принимать решение о типе обработки снимка, применимости алгоритма подавления засветки или выбраковке.

На текущий день реализован некий алгоритм кластеризации данных методом Формальных Элементов. Этот алгоритм взят за основу, как алгоритм, у которого в свойствах не присутствовал недостаток «чувствителен к шумам между кластерами и к перемычкам». Однако сейчас использование алгоритма показывает странные результаты. Первая часть отрабатывает достаточно быстро и хорошо: выделяются таксоны определенного радиуса.

Вторая стадия сейчас проводится методом КНП – кратчайшего незамкнутого пути, термин из теории кластеризации с использованием теории графов. Она подразумевает построение связного графа между центрами всех найденных таксонов и удаление (K-1) наиболее длинных ребер. Таким образом предполагается получить K несвязных графов, вершины каждого из которых, будучи объединены между собой в единый таксон, будут являть искомые кластеры. Тут я столкнулся с тем, что эти таксоны слишком плотно покрывают исследуемое пространство. При условии, что их радиусы одинаковы, расстояние между ними получается примерно одинаковым. Это ведет к тому, что метод КНП недостаточно качественно разделяет исходный связный граф на K искомых. Слишком незначительны оказываются различия в весе ребер графов, рассчитываемые только исходя из расстояния между вершинами. Вероятно, либо следует придумать другой способ задания веса ребра, либо другой способ кластеризации центров полученных таксонов нижнего уровня, либо использовать другой способ разбиения исходного множества точек на таксоны нижнего уровня.

Например, можно использовать разбиение не с постоянным радиусом на пространстве медианы и перцентиля, а с постоянной мощностью кластера.

Алгоритм разбиения на таксоны нижнего уровня предполагает, что движение в сторону наиболее плотного скопления точек происходит за счет пересчета центра масс точек при каждом смещении. Тогда возникает вопрос, в какой момент следует уменьшать (увеличивать) радиус найденного таксона для достижения его требуемой мощности?

Предполагаю, что следует делать так: (модифицированная кластеризация методом ФорЭл)

1. Брать некий начальный радиус
2. Осуществлять передвижение до тех пор, пока таксон не стабилизируется
3. Взять точки, количеством соответствующие требуемой мощности, по принципу наибольшей близости к полученному центру
4. Принять эти точки в качестве очередного таксона, исключить их из числа некластеризованных
5. Повторять до тех пор, пока множество некластеризованных точек не станет пустым.

Таким образом я получаю множество таксонов заданной мощности. Это тоже точки (расположение – центры масс), которые следует кластеризовать. Дальше два пути: рекурсивно использовать кластеризацию методом ФорЭл или подобрать мощность таксонов нижнего уровня, которая позволит на верхнем уровне эффективно использовать КНП.

Попробую второй путь.

2014-05-12

Обнаружено несколько случаев абсолютно некорректной обработки.

Найдена причина: аппроксимация методом градиентного спуска выдает некорректный нефизичный результат (включает NaN). Надо исправить.