

1. Циклические матрицы

В этом разделе рассматриваются операторы свертки на группе Z_n .

Определение. Матрица A называется циклической, если каждый столбец, начиная со второй, получается из предыдущего циклическим сдвигом вниз на один элемент.

$$A = \begin{pmatrix} a_0 & a_{n-1} & \dots & a_1 \\ a_1 & a_0 & \dots & a_2 \\ \dots & \dots & \dots & \dots \\ a_{n-1} & a_{n-2} & \dots & a_0 \end{pmatrix}$$

Определение. Многочлен $a(t) = a_0 + a_1t + a_2t^2 + \dots + a_{n-1}t^{n-1}$ называется многочленом циклической матрицы A .

Определение. Циклическая матрица V с символом $v(t) = t$ называется циклическим сдвигом.

Определение. Если задан полином $f(x) = a_0 + a_1x + a_2x^2 + a_3x^3 + \dots + a_{n-2}x^{n-2} + a_{n-1}x^{n-1}$ и квадратная матрица A , то $f(A) = a_0 \cdot E + a_1 \cdot A + a_2 \cdot A^2 + a_3 \cdot A^3 + \dots + a_{n-2} \cdot A^{n-2} + a_{n-1} \cdot A^{n-1}$.

Теорема. Любая циклическая матрица является многочленом от матрицы V . Точнее $A = a(V)$.

Доказательство. Элементы a_0 находятся только на главной диагонали. Элементы a_1 присутствуют в тех и только в тех местах, где в индексах матрицы V содержатся единицы, элементы a_2 присутствуют в тех и только в тех местах, где находятся единицы в матрице V^2 (иначе говоря, матрице циклического сдвига на два элемента вниз), ..., элементы a_{n-1} присутствуют в тех и только в тех местах, где находятся единицы в матрице V^{n-1} . Таким образом $A = a_0 + a_1V + a_2V^2 + \dots + a_{n-1}V^{n-1}$. Что и требовалось доказать.

Теорема. Пусть $p(x) = p_0 + p_1x + p_1x^2 + \dots + p_nx^n$ – некоторый полином, и X – квадратная матрица с собственными векторами $v_1, v_2, v_3, \dots, v_k$, соответствующими собственным значениям $\lambda_1, \lambda_2, \dots, \lambda_k$. Тогда $v_1, v_2, v_3, \dots, v_k$ являются собственными векторами матрицы $P(X) = p_0 + p_1X + p_1X^2 + \dots + p_nX^n$, соответствующими собственным значениям $p(\lambda_1), p(\lambda_2), p(\lambda_3), \dots, p(\lambda_k)$.

Доказательство. По условию $Xv_i = \lambda_i v_i$. Отсюда $X^m v_i = \lambda_i^m v_i$. Следовательно, $P(X)v_i = (p_0E + p_1X + p_1X^2 + \dots + p_nX^n)v_i = p_0v_i + p_1Xv_i + p_1X^2v_i + \dots + p_nX^nv_i = p_0v_i + p_1\lambda_i v_i + p_1\lambda_i^2 v_i + \dots + p_n\lambda_i^k v_i = (p_0 + p_1\lambda_i + p_1\lambda_i^2 + \dots + p_n\lambda_i^n)v_i = p(\lambda_i)v_i$. Таким образом, $P(X)v_i = p(\lambda_i)v_i$, что и требовалось доказать.

Замечание 1. Любые полиномы от V являются циклическими матрицами, что доказывается схожими с вышеуказанными рассуждениями. Циклическость матрицы X равносильна тому, что X представляется в виде

полинома от V . Циклические матрицы, очевидно, образуют подалгебру алгебры матриц.

Теорема. Векторы

$$v_k = \{1 \quad \varepsilon_k^{-1} \quad \varepsilon_k^{-2} \quad \varepsilon_k^{-3} \quad \dots \quad \varepsilon_k^{-(n-1)}\}$$

являются собственными векторами матрицы A , соответствующим собственным значениям $\alpha(\varepsilon_k)$, где ε_k – корни n -ой степени из единицы.

Теорема. Циклическая матрица A подобна матрице

$$\begin{pmatrix} a(\varepsilon_0) & 0 & 0 \\ 0 & a(\varepsilon_1) & 0 \\ 0 & 0 & a(\varepsilon_{n-1}) \end{pmatrix}.$$

Теорема. Матрица является циклической тогда и только тогда, когда $VX = XV$.

Доказательство. Если X является циклической матрицей, то X является многочленом от V , а значит, коммутирует с V . В обратную сторону, если $V^{-1}XV = X$ то матрица X при циклическом сдвиге вниз и вправо переходит в себя, а отсюда следует, что матрица X является циклической.

Теорема. Если циклическая матрица X обратима, то обратная матрица X^{-1} также является циклической.

Доказательство. Ранее было доказано, что $VX = XV$. Отсюда $X^{-1}V = VX^{-1}$. Следовательно, матрица X^{-1} является циклической.

Теорема. Циклическая матрица обратима тогда и только тогда, когда $a(\varepsilon_k) \neq 0, k = 0, 1, \dots, n-1$.

Доказательство. Обратимость матрицы равносильна тому, что все ее собственные значения не равны нулю.

Определение. Матрица вида $A_k = \frac{1}{k}(E + V^{n-k} + V^{n-k+1} + \dots + V^{n-1})$ называется **матрицей смаза**.

Теорема. Матрица A_k обратима тогда и только тогда, когда $\text{НОД}(k, n) = 1$.

Доказательство. Исследуем обратимость матрицы $kV^k A_k$, которая обратима одновременно с матрицей A_k . $kV^k A_k = E + V^n + V^{n+1} + \dots + V^{n-1}$. Символ этой матрицы равен $1 + t^n + t^{n+1} + \dots + t^{n-1}$, при этом $\varepsilon_0 = 1$ не является корнем символа матрицы. При $t \neq 0$ получим, что символ равен $\frac{t^k - 1}{t - 1}$. ε – корень n -ой степени из единицы, такой что $\varepsilon^k = 1$, существует тогда, когда k и n имеют общий делитель, больший 1, то есть не взаимно простые. Таким образом, согласно ранее доказанной теореме, матрица A_k обратима, когда $(k, n) = 1$, что и требовалось доказать.

Определение. Матрица $A_h = \frac{1}{h}(E + \{h\} \cdot V^{n-[h]-k} + V^{n-[h]} + V^{n-[h]-1} + \dots + V^{n-1})$ называется **матрицей дробного смаза**.

Теорема. Если $\{h\} \neq 0$, то матрица A_h обратима.

2. Восстановление изображения при горизонтальном смазе

Рассмотрим задачу, возникающую при восстановлении смазанного изображения, получаемого горизонтально вращающейся камерой. Эта задача приводит к решению матричного уравнения с циклической матрицей.

В задаче обработки изображений конфликтуют требования к быстродействию, точности и используемой памяти.

Быстродействие восстановления изображений очень важно для задач распознавания образов в режиме реального времени: чем быстрее восстановим изображение, тем больше времени оставим на последующее распознавание.

Обращение к памяти, как правило, является узким местом для большинства прикладных задач. Скорость чтения данных прогрессирует не так быстро, как скорость вычислительных операций. Таким образом, увеличение используемой памяти, как правило, влечет снижение быстродействия.

Решение рассматриваемой задачи сводится к решению системы линейных алгебраических уравнений (СЛАУ) специального вида. В статье [3] приведен алгоритм восстановления изображений для этой задачи, количество операций которого лежит в интервале $(3 \cdot N, 4 \cdot N)$, где N – количество пикселей в изображении. Алгоритм решения СЛАУ в статье [3] является модификацией метода Гаусса. В этом алгоритме, как и в методе Гаусса, матрица системы приводится к треугольному виду. Но при решении системы уравнений с треугольной матрицей, для вычисления одной координаты (последней) неизвестного вектора требуется мало операций, а для вычисления первой координаты неизвестного вектора количество вычислительных операций пропорционально квадрату размерности матрицы. Большой объем вычислений может приводить к накоплению погрешностей округления. Защититься от накопления погрешностей округления можно увеличением длины мантиссы используемых чисел. Но такое действие влечет увеличение памяти промежуточных данных и может повлечь снижение быстродействия.

Итерационный метод Люси-Ричардсона восстанавливает смазанные изображения более общего происхождения, чем в данной работе, работает с матрицей общего вида и требует $C \cdot N^2$ операций. Для восстановления изображений, рассматриваемым в данной работе можно применить преобразование Фурье, что потребует $C \cdot N \cdot \log(N)$ операций. В данной работе рассматриваются алгоритмы сложности $C \cdot N$, которые оказываются возможными за счет специфики матрицы.

При обработке изображений часто возникают задачи, связанные с устранением искажений, вызванных движением снимающей камеры.

В работе [6] рассмотрена модель искажений, вызванная вращением камеры вокруг оптического центра (размытия изображения). Авторы

предлагают свои алгоритмы для решения их задачи, указывая, в частности, на трудоемкость применения метода Люси-Ричардсона.

В работе [7] предложена классификация дрожаний камеры и предложен метод способный устранять размытость изображений, испорченных пространственно-вариантным размытием.

2.1. Влияние погрешностей на качество восстановления изображения.

Влияние вычислительных погрешностей (точность) на качество изображения хорошо видно на представленных ниже рисунках. Смазанность изображения возникает в следствии того, что в процессе экспозиции камера движется относительно снимаемого объекта (или, наоборот, объект движется относительно камеры). Величина смаза определяется скоростью вращения камеры и равна количеству пикселей, на которые сдвигается камера за время экспозиции. На рисунке 1 представлено (очень) смазанное изображение, которое допускает хорошее восстановление. На рисунках 2 и 3 представлены изображения, восстановленные одним и тем же алгоритмом, смазы которых (скорости вращения камеры) отличаются не более, чем на 1 процент, а качество восстановленного изображения отличается очень существенно, что видно невооруженным глазом. Такая разница объясняется различной погрешностью вычислений пикселей восстанавливаемого изображения. Для дальнейшей обработки изображения (например, распознавания образа по шаблону) может быть существенна и такая погрешность, которая не заметна невооруженным глазом.

В [5] выделены 3 источника погрешностей результата вычислений: погрешность метода вычислений, погрешность начальных данных и погрешность машинных округлений.

На рисунках 2 и 3 изображения восстанавливаются одной и той же программой, решающей СЛАУ. При небольшом изменении величины смаза k (которая определяется скоростью вращения камеры) сильно меняется число обусловленности матрицы системы, которое определяет влияние погрешности начальных данных на погрешность результата. Это и объясняет отличие качества представленных восстановленных изображений. Погрешность начальных данных у обоих изображений равна 2^{-16} . Графики зависимости числа обусловленности матрицы СЛАУ от ее размерности при фиксированной величине смаза (скорости вращения камеры) представлены в [3]. В следующем параграфе будет представлен график зависимости числа обусловленности матрицы СЛАУ от величины смаза.

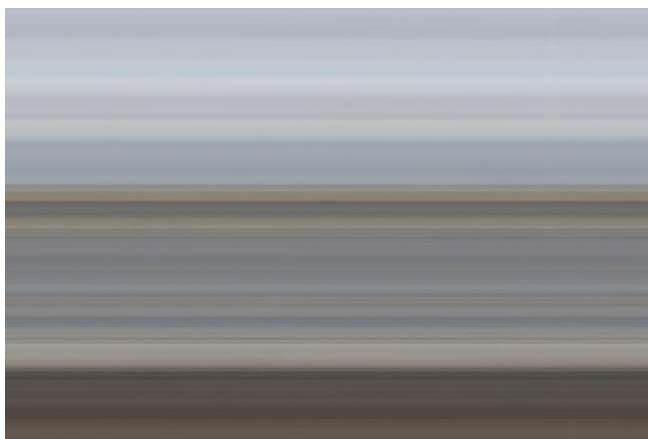


Рисунок. 1. Смаз $k=5563$. Оцифровка 16 бит на цвет. Разрешение 5616x3744

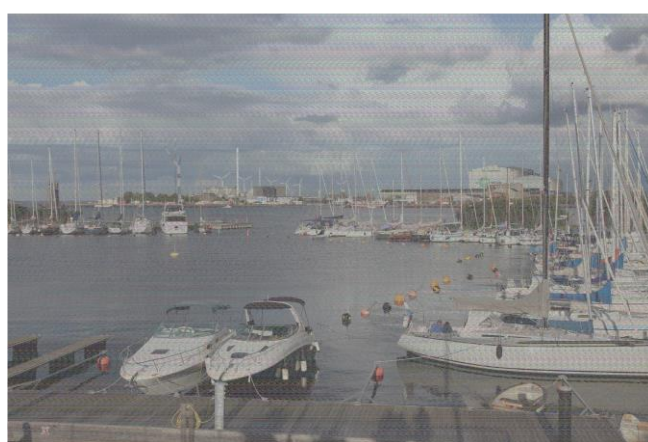


Рисунок. 2. Восстановлено после смаза $k=5563$. Оцифровка 16 бит на цвет. Разрешение 5616x3744.



Рисунок. 3. Восстановлено после смаза $k=5615$. Оцифровка 16 бит на цвет. Разрешение 5616x3744.

2.2. Математическая модель задачи восстановления смазанных цилиндрических панорам и ее особенности.

Рассмотрим идеальную ситуацию, когда смаз на матрице цифровой камеры происходит строго горизонтально

Будем считать, что матрица цифровой камеры имеет разрешение $m \times n$ пикселей, а изображение равномерно сдвигается вправо относительно матрицы на k пикселей. Пусть $\{x_{ij}\} \in M_{m \times N}$ - исходное изображение, которое мы будем представлять как цилиндрическую панораму разрешением $m \times N$. Тогда смазанное изображение будет вычисляться по формуле

$$y_{ij} = \frac{x_{ij} + x_{ij \oplus 1} + \dots + x_{ij \oplus (k-1)}}{k},$$

где $i = 1, 2, \dots, m$, $j = 1, 2, \dots, N$, \oplus - сложение по модулю N . Будем предполагать, что $1 < k < N$. Для восстановления изображения для каждой строки X_i исходного изображения необходимо решить уравнение

$$A(N, k) X_i^T = Y_i^T,$$

где Y_i - соответствующая строка смазанного изображения, $A(N, k)$ - циклическая матрица размеров $N \times N$ вида

$$A(N, k) = \frac{1}{k} \begin{pmatrix} 1 & 1 & 1 & \dots & 1 & 0 & \dots & 0 \\ 0 & 1 & 1 & \dots & 1 & 1 & \dots & 0 \\ 0 & 0 & 1 & \dots & 1 & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 1 & 0 & 0 & \dots & 0 & 0 & \dots & 1 \\ 1 & 1 & 0 & \dots & 0 & 0 & \dots & 1 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 1 & 1 & 1 & \dots & 0 & 0 & 0 & 1 \end{pmatrix} \quad (1),$$

в каждой строке матрицы содержится ровно k единиц.

Отметим некоторые свойства введенной матрицы, описанные в [X].

- Ранг матрицы $A(N, k)$ вычисляется по формуле

$$\text{rang}(A(N, k)) = N - \text{НОД}(N, k) + 1.$$

- Матрица $A(N, k)$ – обратима тогда и только тогда, когда $\text{НОД}(N, k) = 1$.

- Если условие обратимости матрицы выполнено, то ее число обусловленности вычисляется по формуле

$$\text{Cond}(A(N, k)) = \max_{1 \leq l < N} \frac{\left| k \sin \frac{\pi l}{N} \right|}{\left| \sin \frac{\pi k l}{N} \right|}.$$

Число обусловленности показывает, во сколько раз может отличаться погрешность решения СЛАУ от погрешности начальных данных.

Опираясь на представленную формулу зависимости числа обусловленности матрицы СЛАУ от размерности и величины смаза, можно построить график зависимости числа обусловленности этой матрицы от величины смаза k . График зависимости $\text{Cond}(A(N, k))$ от k при фиксированном $N = 720$ показан на рисунке 4.

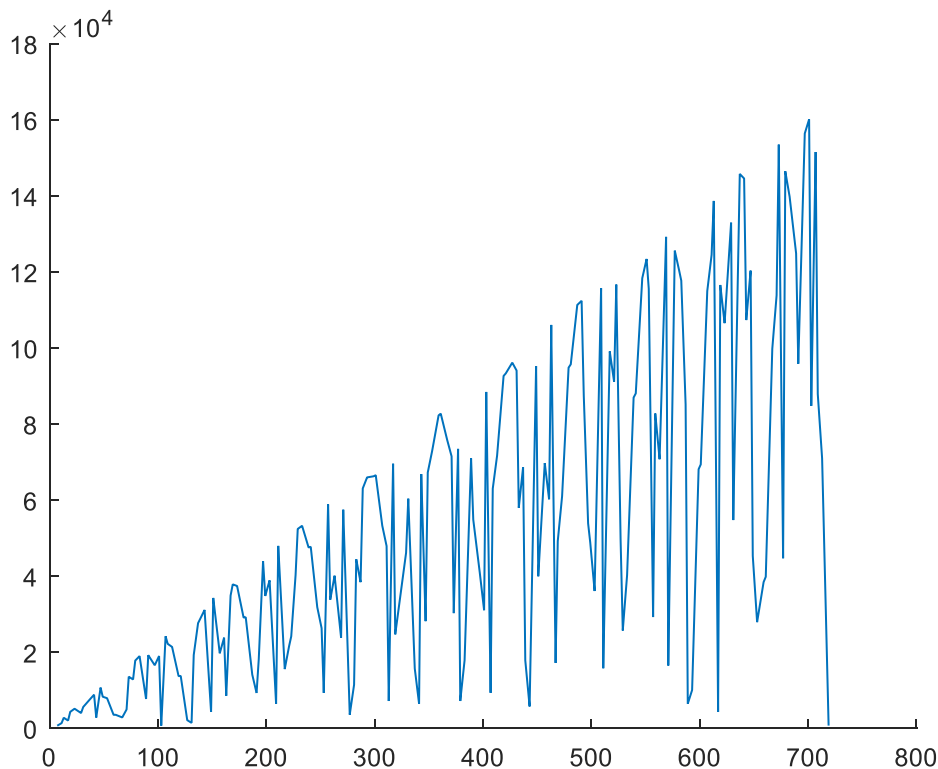


Рисунок 4. Зависимость числа обусловленности матрицы рассматриваемой СЛАУ при $N=720$ от величины смаза k (исключая значения, при которых матрица вырождена).

Из представленного графика видно, что незначительное изменение скорости камеры (величины смаза) может привести к значительным изменениям числа обусловленности матрицы.

На рисунках 2 и 3 изображения заданы с начальной погрешностью 2^{-16} , что намного лучше возможности разрешения бытовых камер 2^{-8} . Промежуточные

вычисления проводились в формате вещественных чисел двойной точности, на процессоре Интел, использующем 80-битный внутренний формат, что практически исключает накопление погрешности округления при обработке изображений размера 1000×1000 .

На рисунке 2 число обусловленности матрицы СЛАУ равно 9940200, а на рисунке 3 число обусловленности соответствующей матрицы равно 5615. Следовательно, погрешность вычисленных пикселей при восстановлении изображений на рисунке 2 и рисунке 3 отличаются в 1770 раз. Это заметно даже при такой высокой точности начального изображения 2^{-16} . Этим объясняется разница в качестве восстановления изображения на рисунках 2 и 3.

Следует отметить, что специализированные вычислители могут работать с числами, имеющими более короткую мантиссу, и при больших объемах вычислений накапливаемые погрешности округления могут превзойти погрешности начальных данных и оказать влияние на восстанавливаемое изображение.

Погрешность вычислений складывается из трех составляющих [5]: погрешность метода, погрешность начальных данных и погрешность машинного округления. В данной работе считаем, что скорость вращения камеры постоянна и заранее известна, следовательно, метод восстановления изображений, как и матрица (1), заданы точно. Зависимость погрешности решения от погрешности начальных данных определяется числом обусловленности матрицы (1) и не зависит от алгоритма решения СЛАУ. Погрешность машинного округления определяется алгоритмом решения, более точно, количеством (и видом) операций, необходимых для вычисления координат результирующего вектора.

3. АЛГОРИТМ ВОССТАНОВЛЕНИЯ СМАЗАННОГО ИЗОБРАЖЕНИЯ, ПОЛУЧЕННОГО ВРАЩАЮЩЕЙСЯ ПОД УГЛОМ К ГОРИЗОНТУ КАМЕРОЙ

Рассмотрим задачу восстановления смазанных изображений, получаемых при равномерном вращении камеры вокруг вертикальной оси, проходящей через оптический центр объектива. Камера делает снимки через равные промежутки времени. Предполагается, что оптическая ось камеры наклонена к плоскости горизонта под некоторым постоянным углом. Снимки получаются смазанными, поскольку сделаны во время движения камеры.

Для многих приложений алгоритм решения задачи восстановления смазанного изображения должен быть достаточно быстрым, чтобы работать в режиме реального времени. Это означает, что время решения задачи должно быть не больше времени получения исходных данных, (последовательности смазанных снимков). В [6] отмечено отставание прогресса в скорости чтения

данных от скорости вычислительных операций. На современных процессорах умножение чисел может быть более чем на порядок быстрее считывания аргументов этого умножения. Для восстановления снимков необходимо, чтобы соседние снимки имели хотя бы минимальное перекрытие. Таким образом, снимков должно быть, с одной стороны, достаточно много, и, с другой стороны, не очень много, чтобы компьютер успевал их обрабатывать.

Рассматриваемую задачу можно отнести к фильтрации. В научной литературе описано много цифровых фильтров, ориентированных на восстановление смазанных изображений, получаемых при движении камеры. Метод Люси-Ричардсона [7], [8], [9] восстанавливает смазанные изображения более общего происхождения, чем в данной работе, работает с матрицей общего вида и требует $C \cdot N^2$ операций, где N – количество пикселей в строке изображения, константа C пропорциональна количеству итераций, которое может иметь порядок нескольких десятков [8]. К близким работам можно отнести методы коррекции изображений в подвижных системах [10], [11]. Известно много методов фильтрации шумов и размытости изображений, которые сводятся к решениям уравнений или вычислениям операторов типа свертки с компактным носителем (опорной областью, функцией рассеяния точки) [12], [13]. К таким же методам относятся и итерационный алгоритм Ван Циттерта [14], фильтр Винера [15] и другие. Большой обзор фильтров представлен в [16]. Рассмотренные в [16] алгоритмы являются алгебраически эквивалентными стандартной реализации фильтра Калмана, но существенно превосходят его по вычислительным свойствам. Быстрый алгоритм восстановления смазанного изображения представлен в статье [17]. Поскольку в основе алгоритма данной работы лежит алгоритм устранения смаза из [1], [2], который превосходит по производительности алгоритм [17], предлагаемый алгоритм также обладает этим свойством. Следует отметить, что метод фильтра Винера не применим к рассматриваемой в данной статье задаче, поскольку этот метод предполагает (после преобразования Фурье) деление на символ исходного оператора свертки, который, как и в [2], может быть вырожден.

Анализ зависимости качества восстановления изображения от параметров съемки представлен в работе [1]. Так как в предлагаемом алгоритме задача восстановления сводится к матричным уравнениям аналогичным уравнениям из [1], то полученные там результаты в полном объеме применимы и к нему.

В работе [1] задача восстановления смазанных снимков сводилась к задаче восстановления смазанной цилиндрической панорамы. В предлагаемой работе восстановление снимков сводится к восстановлению смазанной сферической панорамы.

В [18] приводится классификация панорам, основными классами которых являются цилиндрические, сферические и плоские. Сферические и цилиндрические могут быть как внешними, так и внутренними. В данной работе будут рассматриваться внутренние сферические панорамы. Большинство публикаций о сферических панорамах ориентированы на визуальное рассматривание панорам [18], [19], [20].

Создание панорамы для компьютерного анализа рассмотрено в [21]. В этой работе представлена специальная структура данных Spherex как основа для обработки сферических панорам. Как и в работах, ориентированных на визуальное рассматривание, Spherex строится как склейка нескольких снимков, сделанных фотокамерой в разных направлениях. В работе [21] сформулированы 3 условия, которым должна удовлетворять структура данных для хранения сферической панорамы. В отличие от [21], в данной работе рассматривается не полная сферическая панорама, а панорама, соответствующая полосе сферы. Это позволяет в данной работе использовать двумерный массив для хранения таких панорамных изображений.

Рассматриваемый в данной работе метод восстановления смазанного изображения приводит к задаче удаления смаза на полосе сферической панорамы. Для нее построена математическая модель и разработан алгоритм решения. Программная реализация алгоритма показала высокую эффективность метода

Полученные в данной работе результаты могут быть использованы в робототехнике, в производстве беспилотных летательных аппаратов и других систем мониторинга пространства.

3.1. Сведение задачи восстановления смазанных кадров (снимков) к восстановлению сферических панорам

Перейдем к подробному описанию алгоритма восстановления. Итак, будем рассматривать цифровую фотокамеру, вращающуюся вокруг вертикальной оси, проходящей через оптический центр объектива камеры, с равномерной скоростью. Оптическая ось камеры наклонена под некоторым фиксированным углом к горизонту. Во время выполнения полного оборота камера делает несколько снимков. Снимки получаются смазанными из-за движения камеры.

Воспользуемся простейшей моделью съемки, изображенной на рис.1.

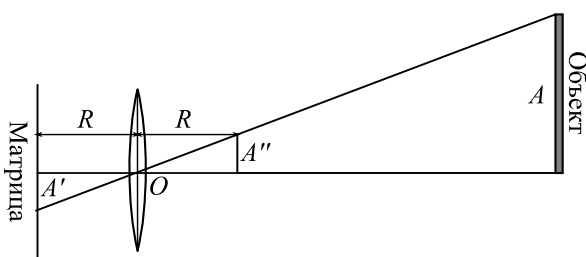


Рис.1. Модель съемки.

Здесь O - оптический центр объектива, R - расстояние от матрицы камеры до оптического центра. Нам удобнее будет работать не с изображением A' объекта съемки на матрице камеры, а с симметричным ему относительно O изображением A'' . Т.е. мы будем считать, что снимаемый объект радиально проектируется (относительно центра O) на плоскость перпендикулярную оптической оси и расположенной между оптическим центром объектива и объектом съемки на расстоянии R от точки O . При вращении камеры вокруг оси, проходящей через точку O смаз на матрице камеры происходит не по прямым, а по кривым второго порядка. Это видно из рис.2.

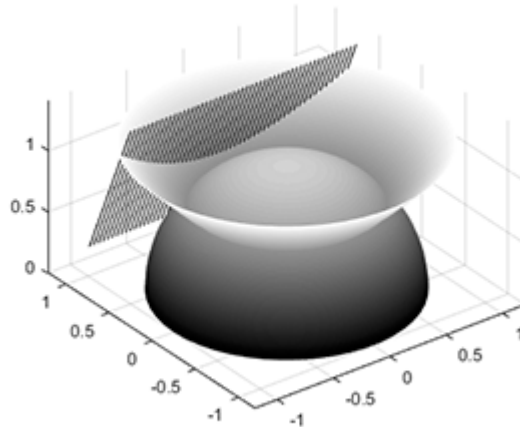


Рис.2. Линии смаза - это линии пересечения конуса и плоскости. Здесь мы считаем, что матрица камеры (на рисунке прямоугольник) неподвижна и касается сферы радиуса R . Произвольный луч с вершиной в центре сферы вращаясь вокруг вертикальной оси описывает конус. Линии смаза - это линии пересечения конуса и плоскости. Образ проекции сферической координатной сетки на касательную плоскость показан на рис.3.

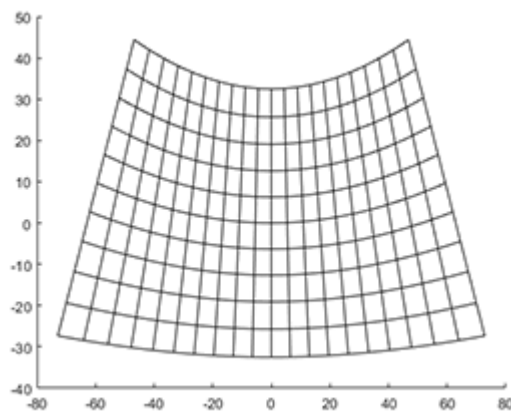


Рис.3. Образ проекции сферической координатной сетки на касательную плоскость.

Смаз происходит по образам параллелей. В нашем случае по кривым второго порядка. Заметим, что чем больше угол наклона оптической оси камеры к горизонту, тем больше кривизна линий смаза, но даже при нулевом угле наклона прямолинейным будет только смаз на горизонтали, проходящей через центр кадра. В тоже время пересечение конуса со сферой является окружностью и смаз на сфере будет проходить по параллелям на сфере. Таким образом, для удаления смаза целесообразно перенести изображение с касательной плоскости на сферу (см. рис.4).

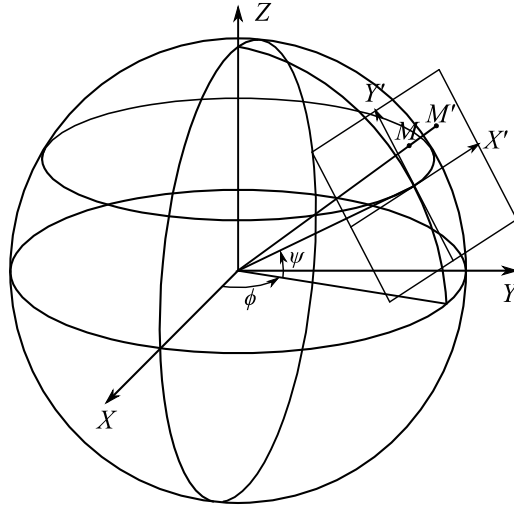


Рис.4. Сопоставление точек касательной плоскости и точек сферы. Здесь R - радиус сферы, (ϕ, ψ) - сферические координаты центра кадра. На самом кадре удобно взять декартову прямоугольную систему координат X', Y' с началом координат в центре кадра. Легко доказать, что приращения сферических координат $(\Delta\phi, \Delta\psi)$ точки M на сфере выражаются через декартовы координаты (x', y') точки M' на кадре по следующим формулам

$$\begin{cases} \Delta\phi = \operatorname{arctg}\left(\frac{x'}{R\cos\psi - y'\sin\psi}\right), \\ \Delta\psi = \arcsin\left(\frac{R\sin\psi + y'\cos\psi}{\sqrt{(x')^2 + (y')^2 + r^2}}\right) - \psi. \end{cases} \quad (1)$$

Заметим, что граница кадра при таком преобразовании переходит в объединение дуг больших кругов на сфере.

В дальнейшем нам понадобятся формулы, выражающие декартовы координаты (x', y') точки M' через приращения сферических координат $(\Delta\phi, \Delta\psi)$ точки M

$$\begin{cases} x' = \lambda R(\cos(\psi_0 + \Delta\psi)\sin\Delta\phi), \\ y' = R\lambda(\sin(\psi_0 + \Delta\psi)\cos\psi_0 - \\ \quad - \cos(\psi_0 + \Delta\psi)\sin\psi_0\cos\Delta\phi), \\ \lambda = (\sin(\psi_0 + \Delta\psi)\sin\psi_0 + \\ \quad + \cos(\psi_0 + \Delta\psi)\cos\psi_0\cos\Delta\phi)^{-1}. \end{cases} \quad (2)$$

Эти формулы, также легко доказываются.

Каждый смазанный кадр из серии, сделанной камерой за один оборот, с помощью формул (1) перенесем на сферу. Фактически мы «вслепую» только по известным углам кадров составляем сферическую панораму (неполную). Пример одного из 10 смазанных кадров приведен на рис.5.



Рис.5. Пример одного из 10 смазанных кадров.

Пример получившейся сферической панорамы приведен на рис.6. Для удобства просмотра сферическая панорама развернута на прямоугольник.

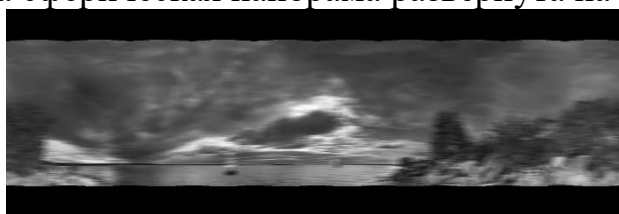


Рис.6. Изображение полосы смазанной сферической панорамы.

Верхняя и нижняя границы панорамы не являются прямыми. Изображение на панораме смазано. Верхняя и нижняя граница панорамы получаются волнообразными. Так как смаз на сфере происходит по параллелям, то однозначно могут быть восстановлены только те из параллелей, которые целиком лежат в панораме. Т.е. узкие полоски, покрывающие волнообразные границы, не восстановимы. Пример перекрытия кадров и восстанавливаемая полоса показаны на рис.7. На этом рисунке показан вид сверху. Восстановлению подлежит только область между двумя концентрическими окружностями.

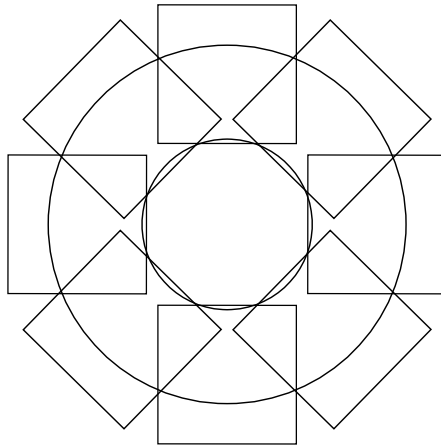


Рис.7. Пример перекрытия кадров.

Между concentрическими окружностями заключена та часть снимков, которая может быть однозначно восстановлена.

Алгоритм восстановления полосы смазанной сферической панорамы будет описан ниже. Пример восстановленной панорамы приведен на рис.8.



Рис.8. Полоса восстановленной сферической панорамы.

На этом рисунке, также, как и на рис.6, показана полоса сферической панорамы, развернутая на прямоугольник. На этом рисунке хорошо видны две узкие полоски вверху и внизу панорамы, где изображение не восстановлено.

3.2. Восстановление смазанной сферической панорамы

Рассмотрим сферу радиуса R . Введем на ней сетку из $2M$ параллелей и N меридиан. Узлы сетки будут иметь следующие сферические координаты (φ_i, ψ_j) , где

$$\varphi_i = \frac{2\pi i}{N}, i = 1, 2, \dots, N,$$

$$\psi_j = \frac{\pi \cdot j}{2M}, j = 0, \pm 1, \pm 2, \dots, \pm M.$$

Пусть на сфере задано некоторое изображение. Т.е. каждому узлу сетки сопоставлено некоторое число x_{ij} - яркость соответствующего пикселя. Таким образом изображение задается матрицей X размеров $N \times (2M + 1)$. В данной работе мы рассматриваем изображения в градациях серого цвета. Цветные изображения задаются тремя матрицами. Работа с ними не отличается от рассматриваемой в этой статье. Исходное изображение равномерно смажем поворотом сферы вокруг вертикальной оси на угол α . Для простоты, чтобы

смаз был целочисленным, будем считать, что $\alpha = \frac{2\pi(k-1)}{N}$, где k - натуральное число. Общий случай не принципиально отличается от рассматриваемого. Пусть смазанное изображение задается матрицей Y . Тогда ее элементы будут вычисляться по формулам

$$y_{i,j} = \frac{1}{k} (x_{i,j} + x_{i+1,j} + \dots + x_{i \oplus (k-1),j}) \quad (3)$$

где $i = 1, 2, \dots, N$, $j = 0, \pm 1, \pm 2, \dots, \pm M$, \oplus - сложение по модулю N . Уравнения смаза (3) можно записать в матричном виде

$$A \cdot X = kY, \quad (4)$$

где A - циклическая матрица размеров $N \times N$ вида

$$A = \begin{pmatrix} 1 & 1 & \dots & 1 & 1 & 0 & 0 & 0 & \dots & 0 & 0 \\ 0 & 1 & \dots & 1 & 1 & 1 & 0 & 0 & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & 1 & 1 & 1 & 1 & 1 & \dots & 0 & 0 \\ 0 & 0 & \dots & 0 & 1 & 1 & 1 & 1 & \dots & 0 & 0 \\ 0 & 0 & \dots & 0 & 0 & 1 & 1 & 1 & \dots & 1 & 0 \\ 0 & 0 & \dots & 0 & 0 & 0 & 1 & 1 & \dots & 1 & 1 \\ 1 & 0 & \dots & 0 & 0 & 0 & 0 & 1 & \dots & 1 & 1 \\ 1 & 1 & \dots & 0 & 0 & 0 & 0 & 0 & \dots & 1 & 1 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 1 & 1 & \dots & 1 & 0 & 0 & 0 & 0 & \dots & 0 & 1 \end{pmatrix} \quad (5)$$

В каждой строке и каждом столбце матрицы (5) содержится ровно k единиц. Для восстановления изображения достаточно решить матричное уравнение (4). Это уравнение подробно исследуется в работах [1-5]. Там предложен эффективный алгоритм для его решения. очевидно, что предложенный алгоритм можно применять и к полосе сферической панорамы, заключенной между двумя параллелями.

4. Описание алгоритма

Исходной информацией для алгоритма является серия смазанных снимков и параметры съемки (угловая скорость вращения камеры, угол наклона оптической оси камеры к плоскости горизонта, промежутки времени, через которые делаются снимки, выдержка, расстояние от оптического центра объектива до матрицы камеры). Алгоритм делится на два этапа.

1. Первый этап. Перенос изображений со смазанных снимков на сферу. Осуществляется с помощью формул (1). При этом надо учитывать, что узлы декартовой решетки на снимке не переходят в узлы сферической решетки. Следовательно, необходимо использовать интерполяцию. Практические вычисления показали, что билинейная интерполяция и, даже, интерполяция по ближайшей точке, дают удовлетворительный результат. После этого на сфере выделяется полоса, полностью покрытая снимками. В результате на первом этапе мы получаем полосу смазанной сферической панорамы.

2. **Второй этап.** Удаление смаза на полосе сферической панорамы с помощью алгоритма, разработанного в работах [2,3]. В результате получается полоса не смазанной сферической панорамы.

Используя формулы (2) легко перенести изображение со сферы на любую касательную плоскость. Т.е. получить снимок аналогичный сделанному неподвижной камерой. Пример такого снимка приведен на рис.9.

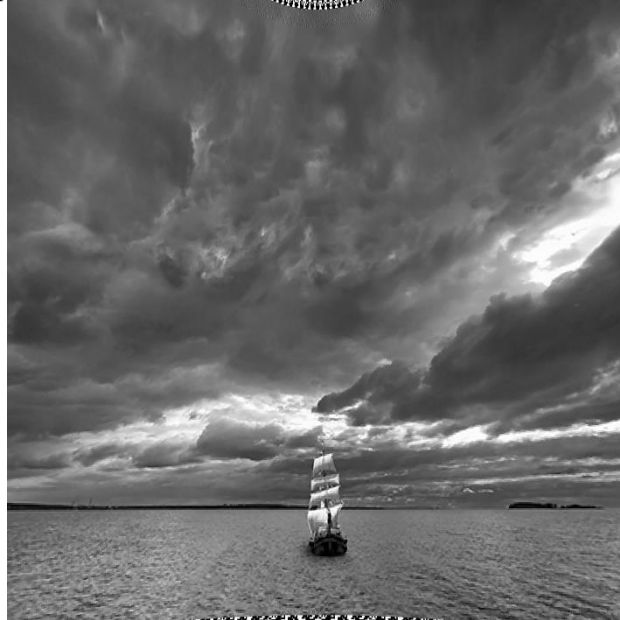


Рис.9. Снимок после удаления смаза.

4. Алгебра Винера

Определение.

$$l_1(\mathbb{Z}) = \{x = \{x_i\}_{i \in \mathbb{Z}} \mid \sum_{i \in \mathbb{Z}} |x_i| < +\infty\}.$$

$$\|x\| = \sum_{i \in \mathbb{Z}} |x_i|$$

Введем операцию свертки двух элементов из $l_1(\mathbb{Z})$: $c = a * b$, где

$$c_k = \sum_{i \in \mathbb{Z}} a_i b_{k-i}$$

Теорема 10. $\|c\| \leq \|a\| \cdot \|b\|$.

Доказательство.

$$\sum_{k \in \mathbb{Z}} \left| \sum_{i \in \mathbb{Z}} a_i b_{k-i} \right| \leq \sum_{k \in \mathbb{Z}} \sum_{i \in \mathbb{Z}} |a_i| \cdot |b_{k-i}| = \|a\| \cdot \|b\|$$

Определение. Для $a \in l_1(\mathbb{Z})$ определим $a(t) = \sum_{i \in \mathbb{Z}} a_i t^i$

Теорема. Операция свертки коммутативна, ассоциативна, билинейна и обладает единичным элементом (им является последовательность со всеми нулями и одной единицей на нулевом месте).

Доказательство. Проверка подстановкой.

Теорема. Функция $a(t)$ непрерывна на единичной окружности, $(\alpha a + \beta b)(t) = \alpha a(t) + \beta b(t)$, $(a * b)(t) = a(t) \cdot b(t)$.

Определение. Линейное пространство $l_1(\mathbb{Z})$ с операцией свертки называется алгеброй Винера и обозначается W .

Теорема. (Теорема Винера). Элемент $a \in W$ обратим, тогда и только тогда, когда $a(t)$ не принимает нулевые значения на единичной окружности.

Определение. Оператор $A_a : l_1(\mathbb{Z}) \rightarrow l_1(\mathbb{Z})$, действующий по правилу $x \mapsto a * x$ называется **оператором свёртки**.

Теорема. Оператор свертки является линейным, причём

$$\|A_\alpha\| = \|\alpha\|.$$

Обозначим как V линейное пространство всех операторов свертки. Оно замкнуто относительно операции композиции. Вектор a – называется ядром оператора A_α . Функция $a(t)$ называется символом оператора A_α .

Теорема. Оператор $A_\alpha \in V$ обратим тогда и только тогда, когда $a(t)$ не принимает нулевое значение на единичной окружности, при этом $A^{-1} \in V$.

Определение. Обозначим через V^- множество операторов свертки с ядрами a такими, что $a_k = 0$ при $k > 0$. А через V^+ множество операторов свертки с ядрами a такими, что $a_k = 0$ при $k < 0$.

Теорема. V^- и V^+ являются подалгебрами алгебры V .

Теорема. Пусть A – оператор свертки с символом $a(t) = t - \alpha$, где $|\alpha| > 1$, тогда $A, A^{-1} \in V^+$.

Доказательство. Так как $A \in V^+$, символ не обращается в нуль на единичной окружности ($|\alpha| > 1$), то также $A^{-1} \in V$.

$\frac{1}{t-\alpha} = -\frac{1}{\alpha} \left[1 + \frac{t}{\alpha} + \left[\frac{t}{\alpha}\right]^2 + \dots \right]$, но $\frac{1}{a(t)}$ имеет ненулевые коэффициенты при неотрицательных t^k , поэтому $A^{-1} \in V^+$.

Теорема. Пусть A – оператор свертки с символом $a(t) = \frac{t-\beta}{t}$, где $|\beta| < 1$. Тогда $A, A^{-1} \in V^-$.

Доказательство. При замене $\frac{1}{t} = z$ имеем $a(z) = 1 - \beta z = -\beta \left[z - \frac{1}{\beta} \right]$.

Далее, рассуждаем аналогично для функции $z - \frac{1}{\beta}$.

Определение. Пусть $a(t)$ – функция, определенная и непрерывная на единичной окружности Γ и $a(t)$ не принимает нулевое значение на Γ . Определим **индекс** функции $a(t)$ формулой

$$\text{Ind}(a) = \frac{1}{2\pi} \Delta_{t \in \Gamma} \arg a(t) = \frac{1}{2\pi} \arg a(e^{j\varphi}) \Big|_{\varphi=0}^{2\pi}$$

Теорема.

$$\text{Ind}(a) = \frac{1}{2\pi j} \Delta_{t \in \Gamma} \ln a(t) = \frac{1}{2\pi j} \ln a(t) \Big|_{\varphi=0}^{2\pi}$$

Теорема. Пусть $a(t)$ – рациональная функция и $a(t) \neq 0$ при $t \in \Gamma$. Пусть количество нулей в единичном круге равно N , а P – количество полюсов в нём же, тогда $\text{Ind}(a) = N - P$.

Теорема. $\text{Ind}(ab) = \text{Ind}(a) + \text{Ind}(b)$

Определение. Оператор U свертки с символом t называется **оператором сдвига**.

Теорема. $U(x)_k = x_{k-1}$

Теорема. Пусть A – оператор свертки с символом $a(t) \neq 0$ на кривой Γ . Тогда существуют оператор A_+ такой, что $A_+, A_+^{-1} \in V^+$, а также оператор A_- такой, что $A_-, A_-^{-1} \in V^-$ такие, что $A = A_- U^{\text{Ind}(a)} A_+$.

Определение. Введем операторы $P_{\pm} : l_1(Z) \rightarrow l_1(Z)$ по правилу

$$P_+(x)_k = \begin{cases} x_k, & k \geq 0 \\ 0, & k < 0 \end{cases}, \quad P_-(x)_k = \begin{cases} x_k, & k < 0 \\ 0, & k \geq 0 \end{cases}$$

Теорема. $A \in V^+$ тогда и только тогда, когда $P_+ A P_+ = P_+ A$.

Теорема. $A \in V^-$ тогда и только тогда, когда $P_- A P_- = P_- A$.

Определение. Пусть $A \in V$, тогда оператор $P_+ A P_+ : \text{Im} P_+ \rightarrow \text{Im} P_+$ называется **оператором Винера-Хонфа**.

Теорема. Если $A \in V$ и $a(t) \neq 0$, на кривой Γ , то $\exists A_{\pm} \in V^{\pm}$ такие, что $A_{\pm}^{-1} \in V^{\pm}$ и $P_+ A P_+ = (P_+ A_- P_+) (P_+ U^{\text{Ind}(a)} P_+) (P_+ A_+ P_+)$.

Теорема. Оператор $P_+ U P_+$ обратим только слева.

Теорема. Оператор $P_+ U^{-1} P_+$ обратим только справа.

Теорема. Пусть A – оператор свертки с символом $a(t) \neq 0$, на кривой Γ , а также $x = \text{Ind}(a)$, тогда:

если $x < 0$, то $P_+ A P_+$ обратим только справа;

если $x = 0$, то $P_+ A P_+$ обратим;

если $x > 0$, то $P_+ A P_+$ обратим только слева.

Теорема (Банаха). Пусть A – обратимый ограниченный линейный оператор, действующий в банаховом пространстве X , тогда любой линейный оператор B такой, что $\|B - A\| < \|A^{-1}\|^{-1}$, обратим.

Теорема. Пусть A – линейный ограниченный оператор, обратимый только справа (слева), тогда в некоторой окрестности A любой оператор обратим только справа (слева).

Теорема. Пусть A – финитный оператор свертки и $a(t_0) = 0$ для некоторого $t_0 \in \Gamma$, тогда оператор P_+AP_+ не может быть обратим ни слева, ни справа.

Теорема. Пусть $A \in V$ и $a(t_0) = 0$ для некоторого $t_0 \in \Gamma$, тогда оператор P_+AP_+ необратим, ни слева, ни справа.

5. Проекционные методы

Пусть X – банахово пространство, а $L(X)$ – множество линейных ограниченных операторов, действующих в X .

Определение. Будем говорить, что A_n **сильно сходится** к A , если $\|A_n x - Ax\| \rightarrow 0$. Обозначается $A_n \rightarrow^S A$.

Теорема (Банаха-Штейнгауза). Если $A_n \rightarrow^S A$, то $\|A_n\|$ ограничены некоторой константой.

Определение. Линейный оператор $P: X \rightarrow X$ называется проектором, если $P^2 = P$.

Определение. Пусть последовательность проекторов P_n сильно сходится к I . Будем говорить, что к оператору A применим проекционный метод по системе P_n , если $\forall b \in X$, уравнение $P_n A P_n = P_n b$ имеет единственное решение, начиная с некоторого значения n , при этом $x_n \in \text{Im} P_n$ и $x_n \rightarrow x_0$, где x_0 – точное решение уравнения $Ax = b$. В этом случае будем писать $A \in \Pi(P_n)$.

Теорема. Обратимый оператор $A \in \Pi(P_n)$ тогда и только тогда, когда, начиная с некоторого значения n , величина $\|(P_n A P_n)^{-1}\|$ ограничена некоторой константой.

Определение. Для множества $U \subset Z^m$ обозначим через P_U оператор

$$(P_U x)_k = \begin{cases} x_k, & k \in U \\ 0, & k \notin U \end{cases}.$$

Теорема. P_U – проектор, причем его норма равна единице, если U не пусто, и равно нулю, если U пусто.

Определение. Для ограниченного оператора A определим функцию $\phi_A(t) = \sup_{\rho(U,V) > t} \|P_U A P_V\|$.

Функция ϕ_A обладает следующими свойствами:

- 1) $\phi_A(t) \leq \|A\|$;
- 2) $\phi_A(t)$ монотонна по t ;
- 3) $\phi_{\alpha A}(t) = |\alpha| \cdot \phi_A(t)$;
- 4) $\phi_{A+B}(t) \leq \phi_A(t) + \phi_B(t)$;
- 5) $\phi_{AB}(t) \leq \|A\| \cdot \phi_B\left(\frac{t}{2}\right) + \|B\| \cdot \phi_A\left(\frac{t}{2}\right)$;

Теорема. Если $A \in V$, то

$$\phi_A(t) \leq \sum_{|k| > t} |a_k|.$$

Отсюда следует, что $\phi_A(t) \rightarrow 0$ при $t \rightarrow +\infty$.

Теорема. Если $a(t) \neq 0$ на кривой Γ , а также $\text{Ind}(a) = 0$, то $\phi_{(P_+ A P_+)^{-1}}(t) \rightarrow 0$ при $t \rightarrow +\infty$.

Теорема. Если $a(t) \neq 0$ на области Γ , а также $\text{Ind}(a) = 0$, то $A \in \Pi(P_n)$.

Литература

1. **Козак, А.В.** Быстрое и точное восстановление смазанного изображения, полученного вращающейся камерой / А.В. Козак, Б.Я. Штейнберг, О.Б. Штейнберг // Труды Международной конференции по программной инженерии СЕЕ-SECR '16. – 2016. – 11. DOI: 10.1145/3022211.3022222
2. **Козак, А.В.** Быстрое восстановление смазанного изображения, полученного горизонтально вращающейся камерой / А.В. Козак, Б.Я. Штейнберг, О.Б. Штейнберг // Компьютерная оптика. – 2018. – Т. 42. – № 6. – С. 1046 – 1053. DOI: 10.18287/2412-6179-2018-42-6-1046-1053.
3. **Козак, А.В.** Уравнение дискретной свёртки с характеристической функцией сегмента и его приложение / А.В. Козак, Б.Я. Штейнберг, О.Б. Штейнберг. – В кн.: Труды научной школы И.Б. Симоненко. Выпуск второй. – Ростов-на-Дону: Изд-во Южного федерального университета, 2015. – С. 157-167.
4. **Козак, А.В.** Оценка погрешностей при решении уравнения свёртки для восстановления смазанных изображений / А.В. Козак, Б.Я. Штейнберг, О.Б. Штейнберг // Тезисы международной конференции «Современные методы и проблемы теории операторов и гармонического анализа и их приложения VI», г. Ростов-на-Дону, 24- 29 апреля 2016 г. – 2016.
5. **Козак, А.В.** Развитие исследований быстрого восстановления смазанного изображения / А.В. Козак, Б.Я. Штейнберг, О.Б. Штейнберг // Тезисы международной конференции «Современные методы и проблемы теории

- операторов и гармонического анализа и их приложения VII», г. Ростов-на-Дону, 23-28 апреля 2017 г. – 2017. – С. 28-29.
6. **Graham, S.L.** Getting up to speed: The future of supercomputing / S.L. Graham, M. Snir, C.A. Patterson. – Washington: National Academies Press, 2005. – 289 p. – ISBN: 978-0-309-09502-0. DOI: 10.17226/11148.
 7. **Lucy, L.B.** An iterative technique for the rectification of observed distributions / L.B. Lucy // The Astronomical Journal. – 1974. – Vol. 79. – P. 745. – DOI: 10.1086/111605.
 8. **Richardson, W.H.** Bayesian-based iterative method of image restoration / W.H. Richardson // Journal of the Optical Society of America. – 1972. – Vol. 62, Issue 1. – P. 55-59. – DOI: 10.1364/JOSA.62.000055.
 9. **Whyte, O.** Non-uniform deblurring for shaken images / O. Whyte, J. Sivic, A. Zisserman, J. Ponce // International Journal of Computer Vision. – 2012. – Vol. 98, Issue 2. – P. 168-186. – DOI: 10.1007/s11263-011-0502-7
 10. **Корнилова, А.** MEMS-датчики в задачах компьютерного зрения: мы их просто недооцениваем [Электронный ресурс] / А. Корнилова, Я. Кириленко // Международная конференция по программной инженерии CEE-SECR '17. – URL: <http://2017.secr.ru/program/submittedpresentations/mems-sensors-in-computer-vision> (дата обращения 8.11.2018).
 11. **Фурсов, В.А.** Интернет-технология коррекции динамических искажений на изображениях в мобильных устройствах [Электронный ресурс] / В.А. Фурсов, П.Ю. Якимов. – 2017. – URL: <http://keldysh.ru/abrau/2017/09.pdf> (дата обращения 8.11.2018). DOI: 10.20948/abrau-2017-09.
 12. **Фурсов, В.А.** Восстановление изображений КИХфильтрами, построенными путём непосредственной идентификации инверсного тракта // Компьютерная оптика. – 1996. – № 16. – С. 103-108.
 13. **Дронникова, С.А.** Улучшение качества изображений при обработке видеокадров с различным временем экспозиции / С.А. Дронникова, И.П. Гуров // Научнотехнический вестник информационных технологий, механики и оптики. – 2017. – Т. 17, № 3. – С. 424-430. – DOI: 10.17586/2226-1494-2017-17-3-424-430.
 14. **Гуров, И.П.** Улучшение качества изображений методом Ван Циттерта / И.П. Гуров, Д.С. Смирнов // Научнотехнический вестник Санкт-Петербургского государственного института точной механики и оптики (технического университета). Информационные технологии, вычислительные и управляющие системы. – 2002. – Вып. 6. – С. 178-182.
 15. **Грузман, И.С.** Цифровая обработка изображений в информационных системах: учеб. пособие / И.С. Грузман, В.С. Киричук, В.П. Косых, Г.И. Перетягин, А.А. Спектор. – Новосибирск: Изд-во НГТУ, 2000. – 168 с.
 16. **Цыганова, Ю.В.** О современных ортогонализированных алгоритмах оптимальной дискретной фильтрации / Ю.В. Цыганова, М.В. Куликова // Вестник ЮУрГУ. Серия «Математическое моделирование и программирование». – 2018. – Т.11. – № 34. – С. 5-30. DOI: 10.14529/mmp180401.

17. **Cho, S.** Fast motion deblurring / S. Cho, S. Lee // ACM Transactions on Graphics. – 2009. – Vol. 28, Issue 5. – 145. – DOI: 10.1145/1618452.1618491.
18. **Smith Chen** // Types of panoramic photography <https://www.picturecorrect.com/tips/types-of-panoramic-photography/> (Дата обращения 19.03.2019)
19. Снимаем панорамы. Часть 1. PROPHOTOS. <https://prophotos.ru/lessons/17978-snimaem-panoramу-chast-1> (Дата обращения 19.03.2019)
20. **Алексеев, В.** Панорамная съемка: основы техники. <https://rosphoto.com/ublogs/panoramnaya-siemka-5411> (Дата обращения 19.03.2019)
21. **Juan David Adarve and Robert Mahony** // Spherpix: a Data Structure for Spherical Image Processing https://www.researchgate.net/publication/311893485_Spherpix_a_Data_Structure_for_Spherical_Image_Processing (Дата обращения 19.03.2019)

Восстановления смазанных изображений.

- При съемке вращающейся камерой возникают смазанные изображения. Например, такое



- Или даже такое



- Для восстановления изображения необходимо решить большую систему линейных алгебраических уравнений специального вида. Результат восстановления может быть такой

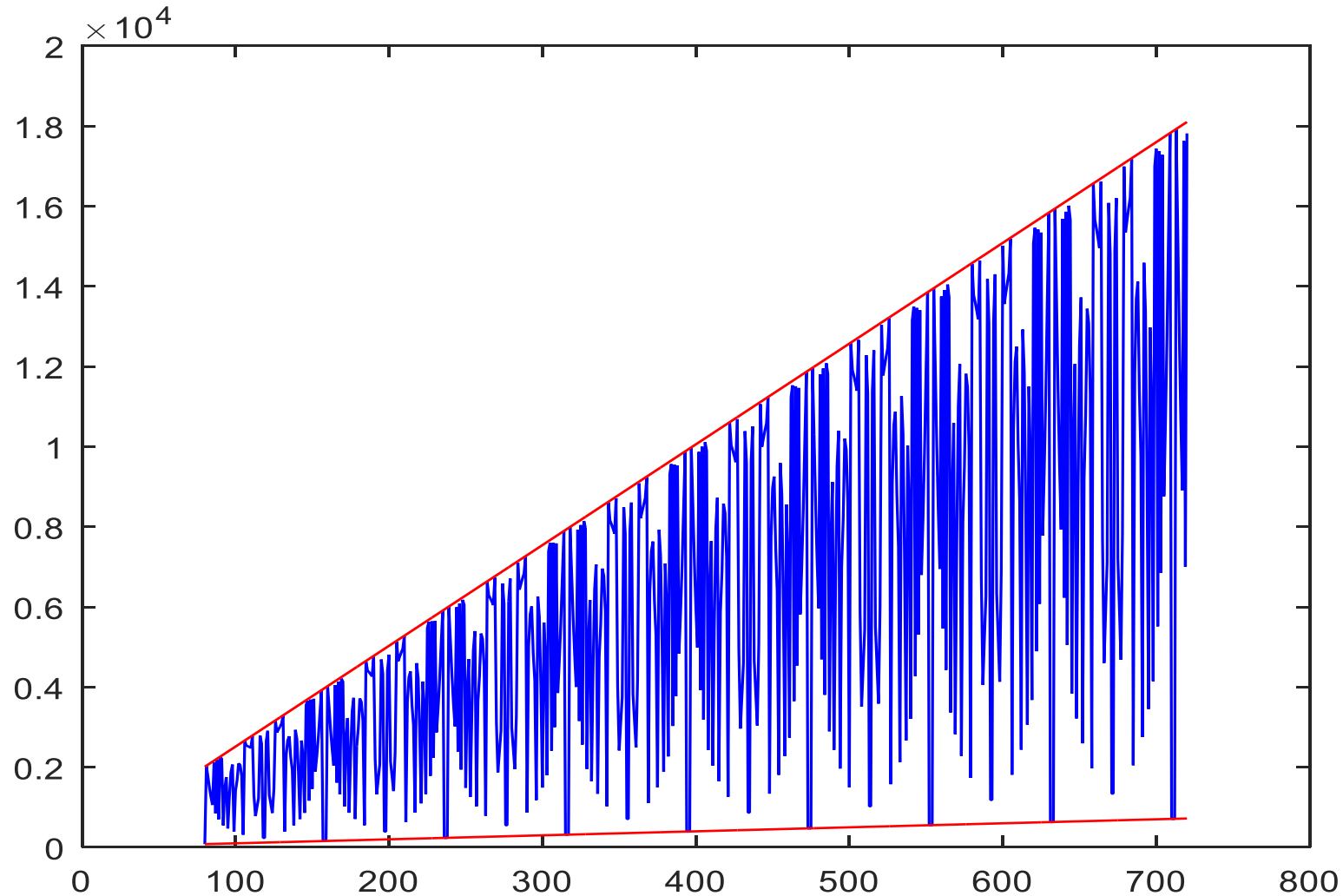


- Или такой



- Качество восстановления зависит от многих факторов:
 1. Точности оцифровки камерой (количество бит на цвет).
 2. Точности вычислений.
 3. Размеров матрицы.
 4. Величины смаза.
 5. Числа обусловленности матрицы смаза.

Зависимость числа обусловленности от размерности матрицы при фиксированной величине смаза (79 пикселей) показана на графике



Уравнение смаза невырождено при

$$d = \text{НОД}(n, h) = 1$$

$$\frac{1}{h} \begin{pmatrix} \underbrace{1 & 1 & 1 & \dots & 1}_{h} & 0 & \dots & 0 \\ 0 & 1 & 1 & \dots & 1 & 1 & \dots & 0 \\ 0 & 0 & 1 & \dots & 1 & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 1 & 0 & 0 & \dots & 0 & 0 & \dots & 1 \\ 1 & 1 & 0 & \dots & 0 & 0 & \dots & 1 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \underbrace{1 & 1 & 1 & \dots & 0 & 0 & 0}_{n} & 1 \end{pmatrix} x = b$$

Рассмотрим случай смаза с вырожденной матрицей. Ранг матрицы: $n - d + 1$; $d = \text{НОД}(n, h)$

$$\frac{1}{h} \begin{pmatrix} \overbrace{1 & 1 & 1 & \dots & 1}^h & 0 & \dots & 0 \\ 0 & 1 & 1 & \dots & 1 & 1 & \dots & 0 \\ 0 & 0 & 1 & \dots & 1 & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 1 & 0 & 0 & \dots & 0 & 0 & \dots & 1 \\ 1 & 1 & 0 & \dots & 0 & 0 & \dots & 1 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \underbrace{1 & 1 & 1 & \dots & 0 & 0 & 0 & 1}_n \end{pmatrix} x = b$$

Задача сводится к СЛАУ размера $(n+1)*n$ с матрицей
 (последнее уравнение взято из исходной СЛАУ)

$$\begin{array}{r}
 100\dots 0-100000 \\
 010\dots 00-10000 \\
 0010\dots 00-1000 \\
 \dots \\
 000\dots 010\dots 0-1 \\
 -1000\dots 010000 \\
 0-100\dots 001000 \\
 \dots \\
 00\dots 0-100\dots 001 \\
 000\dots 0111\dots 111
 \end{array} \quad (3)$$

Последнее уравнение – это уравнение с номером $n-k+1$ исходной системы: $X[n-k+1]+ X[n-k+2]+\dots+ X[n] = H[n-k+1]$

Введение вспомогательных переменных
 $Q[1], Q[2], \dots, Q[d]: Q[1] + Q[2] + \dots + Q[d] = H[n-k+1]$

Заменяем уравнение

$$X[n-k+1] + X[n-k+2] + \dots + X[n] = H[n-k+1]$$

Следующей системой уравнений

$$Q[1] + Q[2] + \dots + Q[d] = H[n-k+1]$$

$$X[n-k+1] + X[n-k+1+d] + \dots = Q[1]$$

$$X[n-k+2] + X[n-k+2+d] + \dots = Q[2]$$

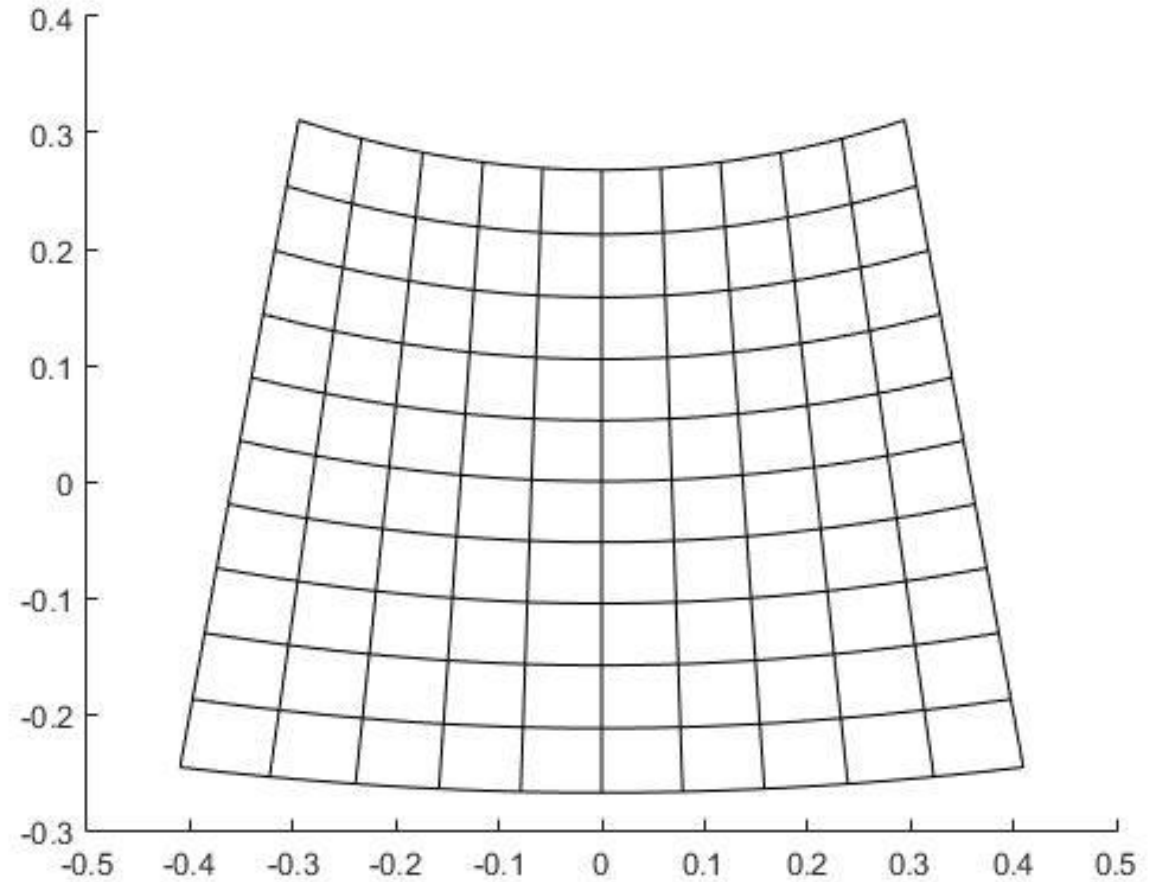
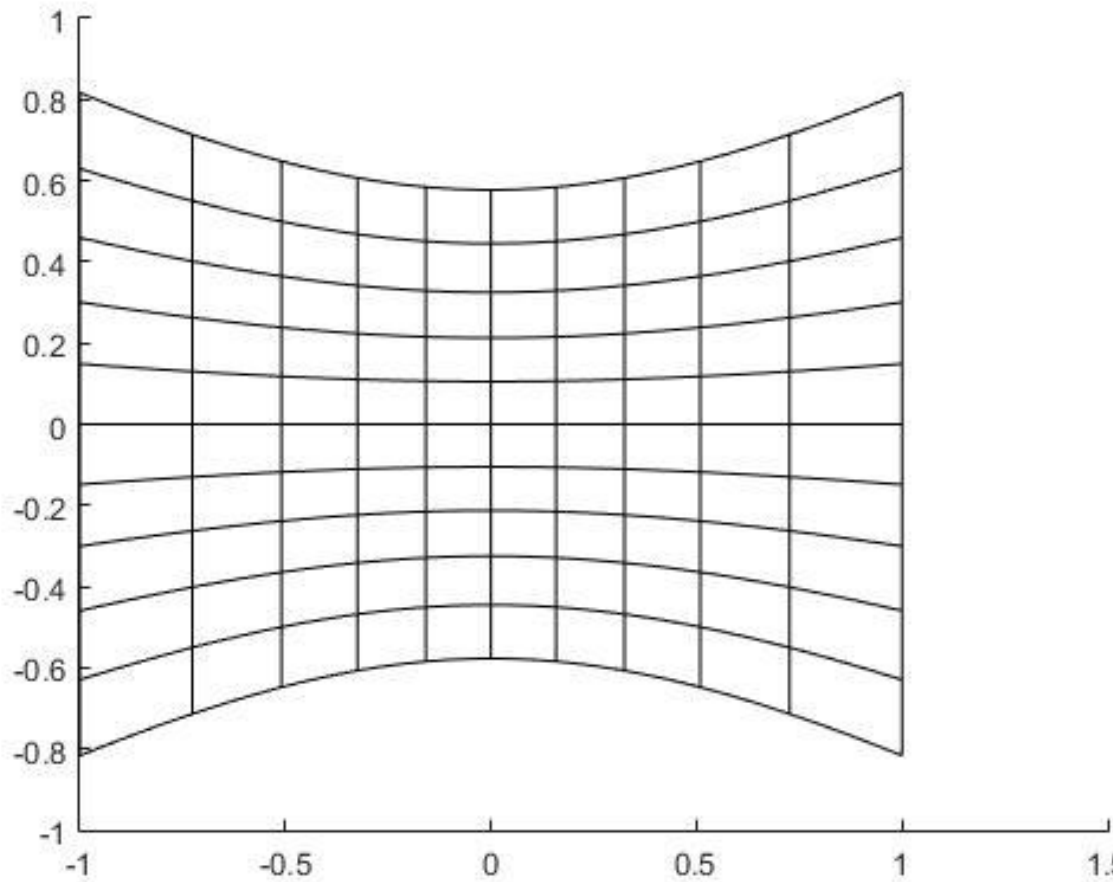
...

$$X[n-k+d] + X[n-k+2*d] + \dots + X[n] = Q[d]$$

У полученных невырожденных СЛАУ размерность в НОД(n, k) раз меньше, чем у исходной. Максимальное число обусловленности таких матриц линейно зависит от размерности. Таким образом, наихудшая точность решения СЛАУ с вырожденной матрицей меньше, чем наихудшая точность решения с невырожденными матрицами того же размера.

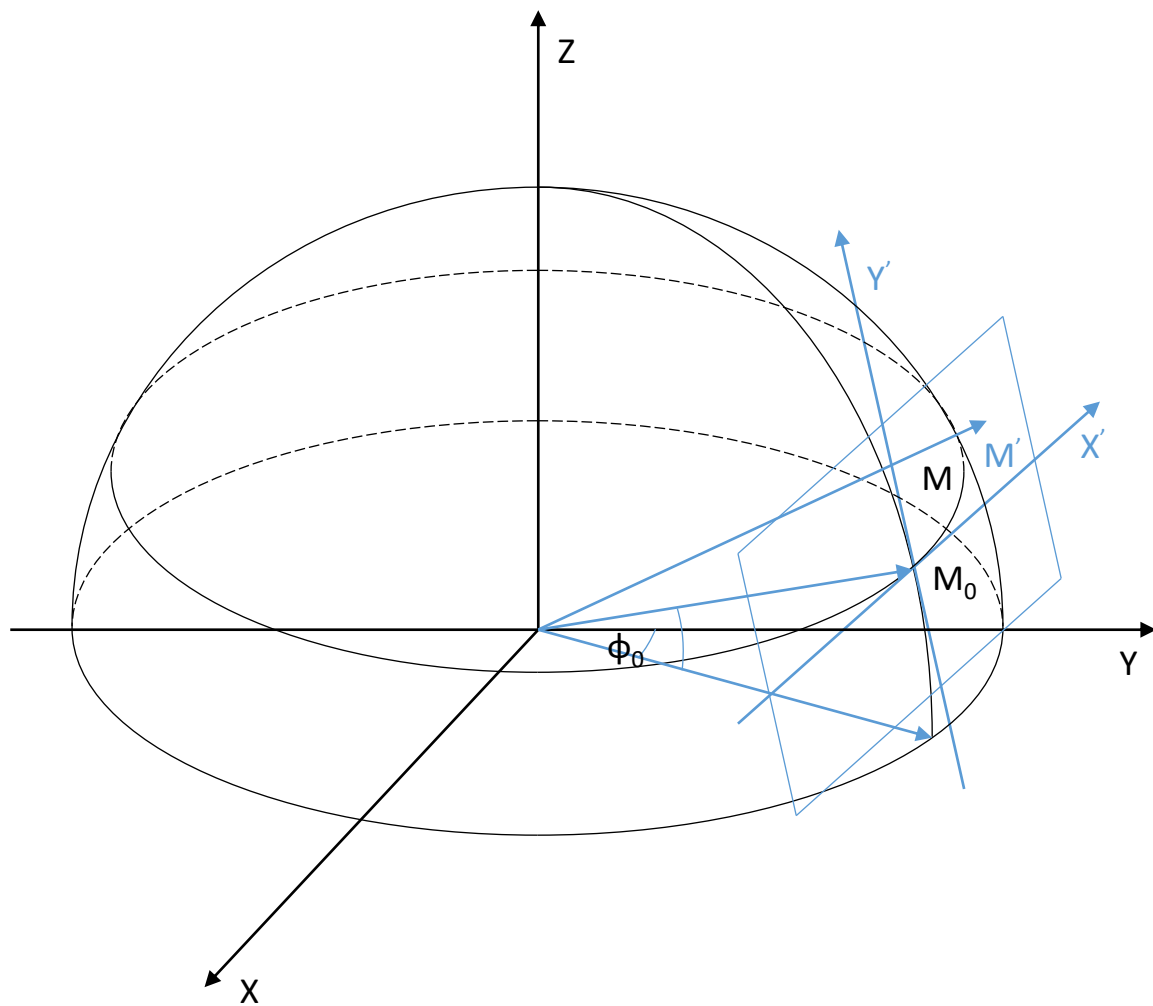
Мы рассмотрели горизонтальный смаз. Так как матрица камеры плоская, то реальный смаз происходит по кривым показанным на следующем слайде.

Сетка смаза при различных углах наклона камеры



- Для компенсации геометрических искажений мы перенесем изображение на сферу. На сфере сдвиг будет происходить по параллелям.

Для компенсации геометрических искажений мы перенесем изображение на сферу. На сфере саз будет происходить по параллелям.



Формулы для компенсации геометрических искажений

$$x' = R\lambda \cos(\varphi_0 + \Delta\varphi) \sin \Delta\phi,$$

$$y' = R\lambda \left(\sin(\varphi_0 + \Delta\varphi) \cos \varphi_0 - \cos(\varphi_0 + \Delta\varphi) \sin \varphi_0 \cos \Delta\phi \right),$$

$$\lambda = \left(\sin(\varphi_0 + \Delta\varphi) \sin \varphi_0 + \cos(\varphi_0 + \Delta\varphi) \cos \varphi_0 \cos \Delta\phi \right)^{-1}$$

R, ϕ_0, φ_0 - сферические координаты центра кадра,

$R, \phi_0 + \Delta\phi, \varphi_0 + \Delta\varphi$ - сферические координаты текущей точки кадра,

(x', y') - декартовы координаты текущей точки кадра.