# Модель ПО и задач

# Анализ

Имеется стационарные поворотная камера. Для неё известны оптические и пространственные характеристики.

По группе последовательных снимков водной местности требуется найти на них объекты и определить их параметры: скорость, траектория (в gps-координатах), размеры объекта. Определить число объектов, которые удалось проанализировать.

Для каждого снимка требуется построить его ортогональную проекцию.

Также для каждого снимка требуется вычислить занимаемую льдом на нём площадь.

**Входные данные**: Группа последовательных снимков, оптические и пространственные характеристики камеры.

**Выходные данные**: Скорость, ускорение, траектория, размер объекта|

сообщение о невозможности анализа (несоответствие снимков друг другу).

**Задачи**:

1. Автоматизировать анализ снимков и расчёты полученных характеристик объектов.
2. В случае ошибок выводить информативные сообщения об ошибках.
3. Программа должна строить ортогональную проекцию выбранного изображения.
4. Для каждого снимка требуется определить занимаемую льдом площадь для выбранного изображения.

Выделенные объекты и их свойства:

* камера
  + оптические характеристики
    - эффективное фокусное расстояние
    - горизонтальный угол обзора
    - вертикальный угол обзора
  + пространственные характеристики
    - высота над уровнем моря
    - угол наклона к горизонту
    - gps-координаты
* снимок
  + пиксельные размеры
    - ширина
    - высота
  + время создания
    - год
    - месяц
    - день
    - час
    - минута
    - секунда
  + положение камеры относительно направления на север
  + объекты
* объект
  + размеры
    - длина
    - высота
    - ширина
  + расстояние до камеры
  + gps-координаты
  + скорость на снимке
  + траектория

# Модель предметной области

# Онтология действительности

сорт объект: {}N каждый объект имеет уникальный идентификатор

сорт снимок: {}N каждый снимок имеет уникальный идентификатор

**сорт** размеры объекта: объект -> размеры

сорт последовательность скоростей объекта: объект -> seq R[0, ∞)

сорт траектория: объект -> seq отрезок

сорт время: (х год, месяц, день, час, минута, секунда)

сорт время получения снимка: снимок -> время

**сорт** объект снимка: снимок -> {} объект

сорт число объектов снимка: снимок -> I [0, 255]

сорт угол положения камеры относительно направления на север: снимок -> угол к направлению на север

**сорт** координата объекта на снимке: (img -> снимок, obj -> объект снимка (img)) -> координата

**сорт** характеристики снимка: снимок -> (х время, оптические характеристики камеры, высота, gps-координаты)

последовательность снимков ≡ seq(снимок)

# сорт расстояние до камеры: (img -> снимок, obj -> объект снимка (img)) -> R[0, ∞)

# Онтология знаний

длина *≡* R (0, 10000]

ширина *≡*  R (0, 10000]

высота *≡*  R (0, 10000]

размеры ≡ (х длина, ширина, высота)

координата *≡* (х I[0, 768], I[0,576])

год *≡* I[2000, 3000]

месяц *≡*  I[1, 12]

день *≡* I[1, 31]

час *≡* I[0, 23]

минута *≡* I[0, 59]

секунда *≡* I[0, 59]

время *≡ (х год, месяц, день, час, минута, секунда)*

отрезок ≡ (х координата, координата)

горизонтальный угол обзора *≡ I(0, 360]*

вертикальный угол обзора *≡ I(0, 360]*

угол наклона к горизонту *≡ I(0, 360]*

угол к направлению на север *≡ I(0, 360]*

эффективное фокусное расстояние*≡ R(0* ∞)

оптические характеристики камеры *≡ (х* горизонтальный угол обзора, вертикальный угол обзора, угол наклона к горизонту, эффективное фокусное расстояние)

# Онтологические соглашения

(t: траектория) (i: I[2, length(t)]) (x1, y1, x2, y2: координата) ∀i, (отр1:π(i, t)) (отр2: next (t, π(i, t))) (отр1: отрезок(x1, y1)) (отр2: отрезок(x2, y2)) => (y2 = x2)

«для каждой траектории любая последовательная пара отрезков имеет общую точку»

∃

# Связь знаний с действительностью

(imgs: **последовательность снимков**) (n: I[2, length(imgs)]) (obj: объект) скорость (imgs, n, obj) = (пройденное объектом расстояние (π(n-1, imgs), π(n, imgs), obj) / (**время получения снимка** (π(n, imgs)) - **время получения снимка** (π(n-1, imgs))))

«скорость объекта obj на n-м снимке из последовательности снимков imgs определяется как пройденное расстояние объектом от своей позиции на n-м снимке до позиции на n-1-м снимке, делённое на разность времени между этими снимками»

(img1, img2: снимок) (obj: объект) пройденное объектом расстояние (img1, img2, obj) = √(расстояние до камеры (img1, obj)^2 + расстояние до камеры (img2, obj)^2 – 2\* расстояние до камеры (img1, obj)\*расстояние до камеры (img2, obj)\*cos(угол положения камеры относительно направления на север (img2) - угол положения камеры относительно направления на север (img1)))

Применяется теорема косинуса: сторона треугольника равна корню из….

(img: снимок) (obj: объект)расстояние до камеры (img, obj)

# Постановка прикладных задач

Дано:

* последовательность снимков
* высота камеры
* оптические характеристики камеры

Найти:

* число объектов
* скорость каждого объекта
* расстояние до каждого объекта
* направление каждого объекта на каждом снимке
* географические координаты каждого объекта
* процентное соотношение льда и остальной поверхности на каждом снимке
* ортогональную проекцию местности на каждом снимке

1. Пользователь загружает в систему снимки для анализа.
2. Если снимки удовлетворяют условиям (по метаданным они являются последовательными), то система применяет методы улучшения изображений снимков: <перечисление методов>.
3. Система определяет число объектов.
4. Система определяет приблизительные размеры объектов.
5. Система определяет скорость каждого объекта.
6. Система определяет траекторию каждого объекта.
7. Система определяет GPS-координаты каждого объекта.
8. <тут написать про лёд>
9. <система выводит полученные данные на экран>
10. Если пользователь запрашивает горизонтальный снимок определённого изображения, то система строит ортогональную проекцию этого изображения.

# Методы решения

мат формулы