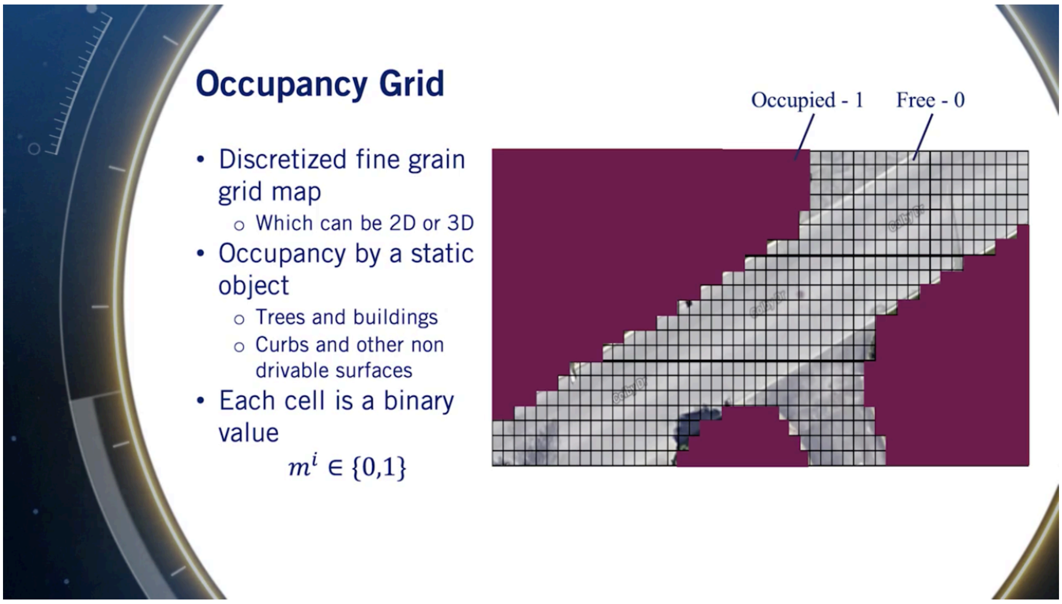
Module 2: Mapping for Planning

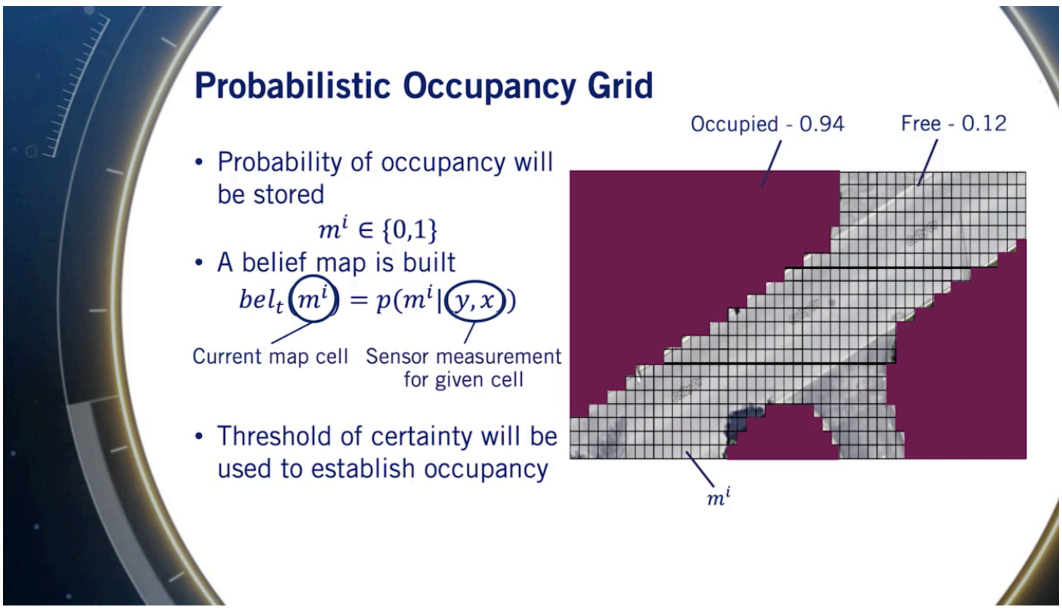
The occupancy grid map and the high-definition road map.

Lesson 1: Occupancy Grids

内容
- Define occupancy grid.
  - Creation of occupancy grid using autonomous car sensors.
- Noise inherent to measurement data used to construct occupancy grid.
- Handling noisy data by using Bayesian updates.



Occupancy Grid
- Discretized find grain grid map.
  - Which can be 2D or 3D.
- Occupancy by a static object.
  - Trees and buildings.
  - Curbs and other non drivable surfaces.
- Each cell is a binary value: $m^i \in \{0,1\}$.
- Assumptions of Occupancy Grid.



- Static environment.
- Independence of each cell.
- Known vehicle state at each time step.

LIDAR Data Filtering（このfilteringをlidar point cloudを表示するprogramに入れたら、全部point cloudが表示できるかも！）

- Several components of the LIDAR data need to be filtered out before this data can be used to construct an occupancy grid.
- The first step is to filter all LIDAR points which comprise the ground plane.
- Next, all points which appear above the highest point of the vehicle are also filtered out.
    - だから上記のoccupancy grid図に若干木の枝が出ているところもfreeになれるんだ！
    - This set of LIDAR points can be ignored as they will not impede the progression of the autonomous vehicle.
- Finally, all non-static objects which had been captured by the LIDAR need to be removed.
- Projection onto a 2D plane.
- Range Sensorになっちゃう。
    - 2D range sensor measuring distance to static objects.
- 残る課題：LIDAR Data Noise.
    - 解決策：Probabilistic Occupancy Grid.
        - Probability of occupancy will be stored.
        - A belief map is built: $bel_t(m^i) = p(m^i \,|\, (y, x))$.
        - Threshold of certainty will be used to establish occupancy.

Bayesian Update of the Occupancy Grid

- To improve robustness, multiple timesteps are used to produce the current map: $bel_t(m^i) = p(m^i \,|\, (y, x)_{1:t})$.
    - m^iの値は結局binaryです。つまりoccupiedかどうか。
- Bayes's theorem is applied at each update step for each cell: $bel_t(m^i) = \eta p(y_t \,|\, m^i) bel_{t-1}(m^i)$.
    - $p(y_t \,|\, m^i)$: measurement model.
        - the probability of getting a particular measurement given a cell m^i is occupied.
        - y_t: the sensor measurement.
    - We rely on the Markov assumption, that all necessary information for estimating cell occupancy is captured in the belief map at each time step.
    - η: normalizing constant.

Lesson 2: Populating Occupancy Grids from LIDAR Scan Data (Part 1)

単語

- Logit: In statistics, the logit function or the log-odds is the logarithm of the odds $\dfrac{p}{1-p}$ where p is probability. $logit(p) = log\left(\dfrac{p}{1-p}\right)$.
    - It is a type of function that creates a map of probability values from $[0,1)$ to $(-\infty, +\infty)$.
    - It is the inverse of the sigmoidal "logistic" function or logistic transform: $logit^{-1}(\alpha) = logistic(\alpha) = \dfrac{1}{1+exp(-\alpha)} = \dfrac{exp(\alpha)}{exp(\alpha)+1}$.
- Rounding: a process in which a number is approximated as the closest number that can be expressed using the number of bits or digits available.

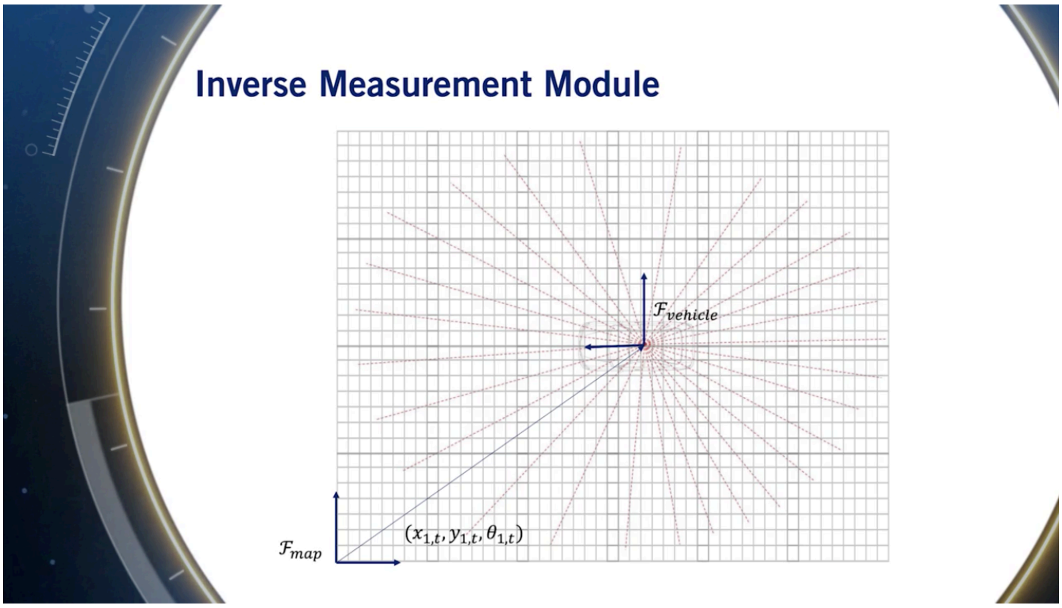
内容
- Issue with the Bayesian Probability Update.
- A solution utilizing log odds.
- Bayesian log odds update derivation.
  - required to update the belief map.

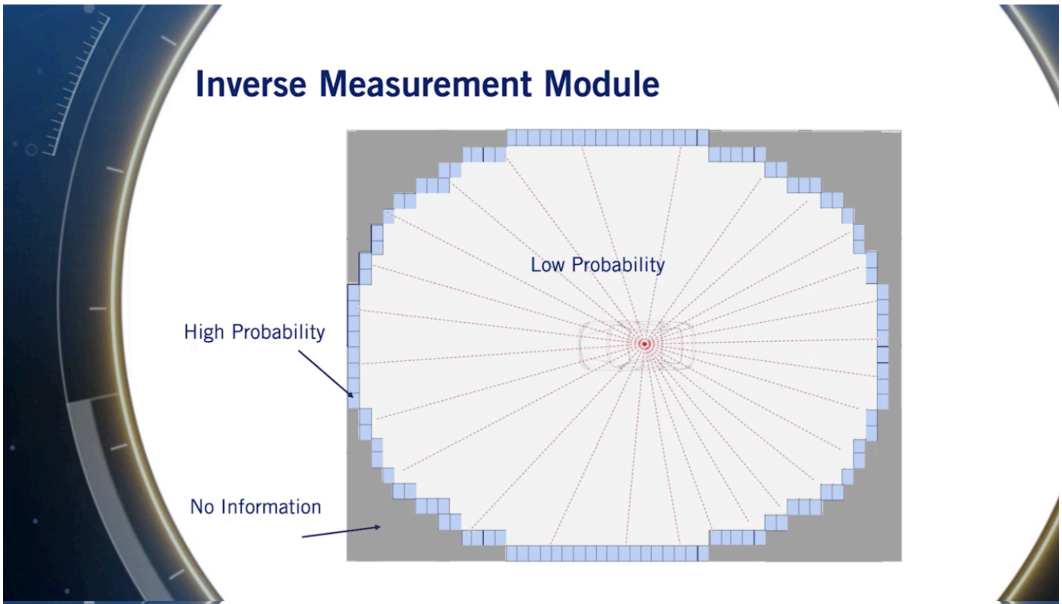
Issue with Standard Bayesian Update
- 例：Update a single unoccupied grid cell.
  - $p(y_t|m) = 0.000012$, $bel_{t-1}(m) = 0.000638$, then $bel_t(m) = 0.000000008$.
- Multiplication of numbers close to zero is hard for computers.
  - Multiplication of floating-point numbers on a computer can lead to significant rounding error when multiplying small numbers, which in turn can lead to instability in the estimate of the probabilities.
- Store the log odds ratio rather than probability: $bel_t(m) \rightarrow (-\infty, \infty)$ by $log\left(\frac{p}{1-p}\right)$.
- 逆：$p = \frac{e^{logit(p)}}{1 + e^{logit(p)}}$。つまりlogistic function。

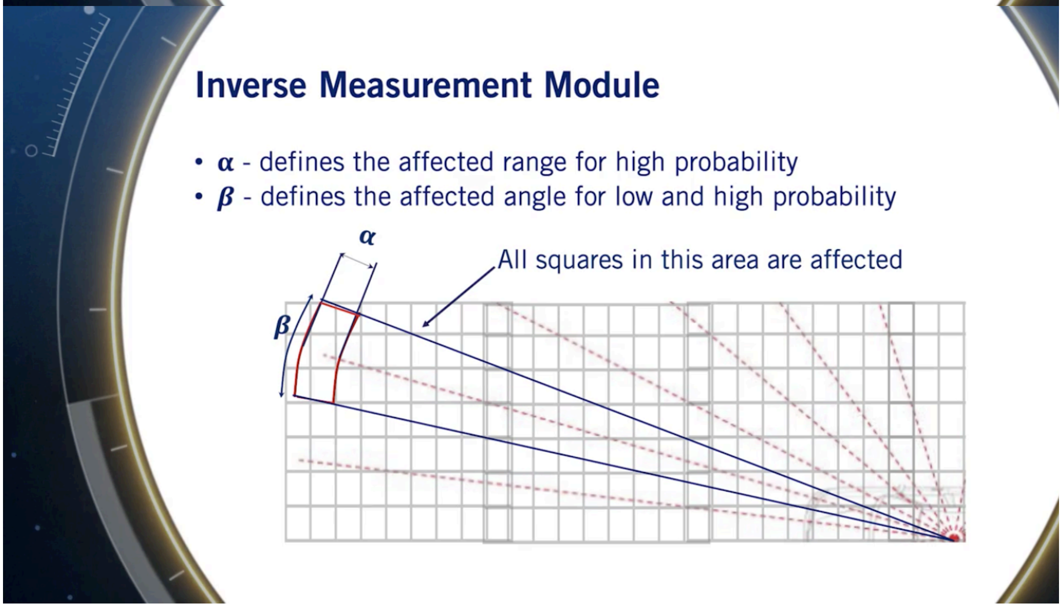
Bayesian Log Odds Single Cell Update Derivation（大事）
- Applying Bayes' rule: $p(m^i|y_{1:t}) = \frac{p(y_t|y_{1:t-1}, m^i)p(m^i|y_{1:t-1})}{p(y_t|y_{1:t-1})}$.
-
$$
\begin{aligned}
p(m^i|y_{1:t}) &= \frac{p(m^i, y_{1:t})}{p(y_{1:t})} \\
&= \frac{p(y_t|y_{1:t-1}, m^i)p(y_{1:t-1}, m^i)}{p(y_t|y_{1:t-1})p(y_{1:t-1})} \\
&= \frac{p(y_t|y_{1:t-1}, m^i)p(m^i|y_{1:t-1})p(y_{1:t-1})}{p(y_t|y_{1:t-1})p(y_{1:t-1})}
\end{aligned}
.
$$
  - Pulling out current measurement y_t from past measurements $y_{1:t-1}$.
    - 理由： we would like to update the occupancy grid with only the most recent sensor measurement (y_t) rather than storing all measurements and applying them again every time.
- Applying the Markov assumption: $p(m^i|y_{1:t}) = \frac{p(y_t|m^i)p(m^i|y_{1:t-1})}{p(y_t|y_{1:t-1})}$.
  - Markov assumption ensures that the current measurement is independent of previous measurements if the map state m^i is known.
- Applying Bayes' rule to measurement model: $p(y_t|m^i) = \frac{p(m^i|y_t)p(y_t)}{p(m^i)}$.
- Yields: $p(m^i|y_{1:t}) = \frac{p(m^i|y_t)p(y_t)p(m^i|y_{1:t-1})}{p(m^i)p(y_t|y_{1:t-1})}$.
- Denominator: $1 - p$.
  - $p(\neg m^i|y_{1:t}) = 1 - p(m^i|y_{1:t}) = \frac{p(\neg m^i|y_t)p(y_t)p(\neg m^i|y_{1:t-1})}{p(\neg m^i)p(y_t|y_{1:t-1})}$.
  - Slidesの式は間違っている（でもその後のslideに修正された）。証拠：http://ais.informatik.uni-freiburg.de/teaching/ws17/mapping/pdf/slam10-gridmaps.pdf
- Logit function: $logit(p) = log\left(\frac{p}{1-p}\right)$.

- $$\frac{p(m^i|y_{1:t})}{p(\neg m^i|y_{1:t})} = \frac{\frac{p(m^i|y_t)p(y_t)p(m^i|y_{1:t-1})}{p(m^i)p(y_t|y_{1:t-1})}}{\frac{p(\neg m^i|y_t)p(y_t)p(\neg m^i|y_{1:t-1})}{p(\neg m^i)p(y_t|y_{1:t-1})}}.$$
- Simplifying like terms results in:
$$\frac{p(m^i|y_{1:t})}{p(\neg m^i|y_{1:t})} = \frac{\frac{p(m^i|y_t)p(m^i|y_{1:t-1})}{p(m^i)}}{\frac{p(\neg m^i|y_t)p(\neg m^i|y_{1:t-1})}{p(\neg m^i)}} = \frac{p(m^i|y_t)p(\neg m^i)p(m^i|y_{1:t-1})}{p(\neg m^i|y_t)p(m^i)p(\neg m^i|y_{1:t-1})}.$$
- Can rewrite by taking $\neg p$ to $1-p$:
$$\frac{p(m^i|y_{1:t})}{p(\neg m^i|y_{1:t})} = \frac{p(m^i|y_t)\left(1-p(m^i)\right)p(m^i|y_{1:t-1})}{\left(1-p(m^i|y_t)\right)p(m^i)\left(1-p(m^i|y_{1:t-1})\right)}.$$
  - 3つlogitになっている!
- Finally, taking the log:
$$logit\left(p(m^i|y_{1:t})\right) = logit\left(p(m^i|y_t)\right) + logit\left(p(m^i|y_{1:t-1})\right) - logit\left(p(m^i)\right).$$
  - 更に簡単な書き方: $l_{t,i} = logit\left(p(m^i|y_t)\right) + l_{t-1,i} - l_{0,i}$.
  - $l_{0,i}$の書き方は面白い! priorの考え方!
  - Kalman Filterは本当大事ですね! measurementsを使う場合はほとんどKalman Filterを使わなきゃあかんでしょう!

Bayesian log odds Update
$$l_{t,i} = logit\left(p(m^i|y_t)\right) + l_{t-1,i} - l_{0,i}$$
- $p(m^i|y_t)$: Inverse Measurement Model.
  - measurement modelは$p(y_t|m^i)$だから!
- $l_{t-1,i}$: Previous Belief at cell i.
- $l_{0,i}$: Initial Belief at time zero at cell i.
  - The initial belief represents the baseline belief that a grid cell is occupied, which is usually set to 50% uniformly.
  - It shows up in this equation at every time step, which is a bit surprising but is simply a result of the derivation and adjusts the addition of the first two terms to ensure the updated belief is consistent with the log odds form.
- メリット:
  - Numerically stable.
    - due to logit.







  - Computationally efficient.

Lesson 2: Populating Occupancy Grids from LIDAR Scan Data (Part 2)

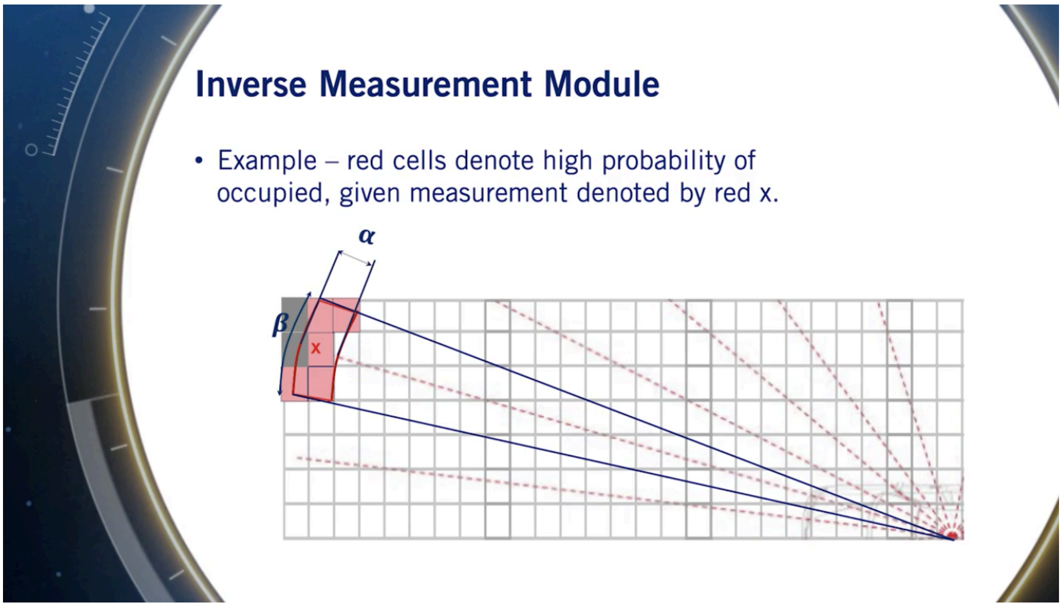
内容

- Create a simple Inverse Measurement Model.
  - つまりmeasurementからgrid cellがoccupied確率を出せるmodel!
- An improvement using Bresenham line algorithm.
  - ray tracing which will significantly reduce computational requirements for the inverse measurement model.
- 今回のLessonはちょっとミスが多いと思う。

Inverse Measurement Module - To be improved

- Scanner bearing: $\phi^s = [-\phi^s_{max} \ \ldots \ \phi^s_{max}]$, $\phi^s_j \in \phi^s$.
- Scanner ranges: $r^s = [r^s_1 \ \ldots \ r^s_j]$, $r^s_j \in [0, r^s_{max}]$.

- Most lidar today will also return a no echo signal, if a particular beam does not return an echo, indicating the absence of any object within range in that direction.
- We will make an assumption here that the entire lidar scan measurements from a single rotation of the sensor is measured at the same instant in time.
  - 本当は車の動きを考えてmeasurementsを調整しないといけない。
- We construct a temporary occupancy grid that encompasses the maximum range of the beams in all directions.
- In practice, this occupancy grid frame is set to the vehicle frame and the map is transformed at each step based on our state estimates.
- current LiDAR dataを基づいて、3つdistinct areas of the measurement grid.
  - No information.
    - None of the lidar beams had been able to reach.
  - Low probability.
    - all the beams have passed through this area without encountering anything.
  - High probability.
    - in which lidar has come into contact with an object and has returned a non maximum range value.
- To translate these areas onto the measurement grid, each grid square will be assigned a range and bearing relative to the vehicle's current location.



- Relative range: $r^i = \sqrt{\left(m_x^i - x_{1,t}\right)^2 + \left(m_y^i - x_{2,t}\right)^2}$.
  - $x_{1,t}, x_{2,t}$ because vehicle's current location changes.
  - m_x^i, m_y^i are coordinates of the center of the grid cell.
- Relative bearing: $\phi^i = tan^{-1}\left(\frac{m_y^i - x_{2,t}}{m_x^i - x_{1,t}}\right) - x_{3,t}$.
- Closest relative bearing: $k = argmin\left(|\phi^i - \phi_j^s|\right)$.
  - 一番近いlidarの1本beam。
  - for each cell, associate the most relevant lidar beam.
- α: defines the affected range for high probability.

- β: defines the affected angle for low and high probability.

Inverse Measurement Module - Algorithm
- このページの内容は講師が教えた内容と全然違うと思うので、直接に例へ進む。（上記Screenshotを参考）
  - 大体のやり方は簡単。lidarのmeasurement (red x)からα, βによって決まった範囲内にgridの中心があったら、このgridはHigh Probability。
  - 角度がβ範囲内だが、距離は$measurement - \alpha/2$以下だったら、Low Probability。
- 例えば、a lidar scan returns a range to an object at the location marked by a red x.
  - red cells denote high probability of occupied.
- 宿題にこれを実現する関数inverse_scannerがあって、正しい内容が入っている：
- # If the range is greater than the maximum sensor range, or behind our range
-         # measurement, or is outside of the field of view of the sensor, then no
-         # new information is available.
-         if (r > min(rmax, meas_r[k] + alpha / 2.0)) or (abs(phi - meas_phi[k]) > beta / 2.0):
-             m[i, j] = 0.5
-
-         # If the range measurement lied within this cell, it is likely to be an object.
-         elif (meas_r[k] < rmax) and (abs(r - meas_r[k]) < alpha / 2.0):
-             m[i, j] = 0.7
-
-         # If the cell is in front of the range measurement, it is likely to be empty.
-         elif r < meas_r[k]:
-             m[i, j] = 0.3
- 最後のelif文は実2つ場合があって：
  - meas_r[k] >= rmax && r < meas_r[k]
    - つまりobstacleが存在しない場合、この方向の有効範囲内のgrid cellは全部low probability。
  - meas_r[k] < rmax && r < meas_r[k] - alpha / 2.
    - high probabilityのelse ifなので、結局r < meas_r[k] - alpha/2。
    - つまりobstacleが存在する場合、measurement-alpha/2より手前以内はobstacleがないよ！

Inverse Measurement Module With Ray Tracing
- Ray tracing algorithm using Bresenham's line algorithm.
  - originally designed to efficiently solve the line plotting problem for displays and printing on the available hardware of the day.
  - rasterized line algorithm.
  - Uses very cheap fixed point operations for fast calculations.
- Perform update on each beam from the LIDAR rather than each cell on the grid.
  - By ray tracing along the beams of the lidar scan, we reduce the number of cells that need to be processed and identify them more quickly relying on integer addition, subtraction and bit shifting to move through the grid along the lidar beams.

Lesson 3: Occupancy Grid Updates for Self-Driving Cars

単語
- Concave: A surface that is concave inward in the middle.
- Drainage: Drainage is the system or process by which water or other liquids are drained from a place.

内容
- Requirement for converting 3D lidar data to 2D data suitable to be used by occupancy grid.
  - Set of filters required for 3D lidar data.

- 3D to 2D projection.
- Tuning the occupancy grid for the task of autonomous driving.

Filtering of 3D LIDAR（Lesson 1に既にあった）（これはlidar point cloudのpublishに使うべき）
無理だ、Ground PlaneやDynamic Objectsのfilteringは全部カメラのperceptionに依頼する。
- Downsample.
  - 理由：make update operations run in real-time.
- Objects above car height.
- Ground Plane.
  - This drivable surface comes from the perception modules.
- Dynamic Objects.
  - これもrely on perception.

Downsampling
- Up to ~1.2 million points per second.
- Removal of redundant points.
  - redundantというのは、pointsの座標が同じということじゃなく、同じobject上のpointsという意味！
- Improves computation.
- Downsamplingはavailable in Point Cloud Library and OpenCV.

Removing of overhanging objects
- This filter usually presumes a flat ground plane, and so should be aware that this is a dangerous assumption to apply blindly.
  - accelerometerと同じような危なさ。

Removal of ground plane
- Due to the nature of the lidar scan, many of the concentric circles are due to the lidar scan hitting the drivable surface.
- やる理由（大事）：If the ground plane is not removed, the occupancy grid might have artifacts which can result in deadlock in which the car cannot continue driving as it believes the roadway is blocked.
- Difficult to estimate due to several complications.
  - これはperceptionのdrivable planeの予測とはどんな関係？？
  - Differing road geometries.
    - variable concavity for drainage.
    - different slopes and bank angles.
    - curvatures.
  - Curbs, lane boundaries.
    - Curbs have different heights at different locations and road boundaries are not always clearly defined in lidar data.
    - parts of the curbs or non-drivable areas being removed as parts of the ground plane.
  - Don’t want to miss small objects.
- The best approach to dealing with this sort of issue is to take advantage of vision, and deep neural networks through semantic segmentation!（上記質問の答え、関係はperceptionがまさに解決方法だ！）

Ground plane Classification
- Utilize segmentation to remove points of road elements.
- The task then becomes a mapping of the drivable surface detected in the vision data to the lidar point cloud, masking out all those points that fall within the projected boundaries of the drivable surface.
- Keep points from no drivable surfaces.

Removal of Dynamic Objects
- またPerceptionに依頼する！

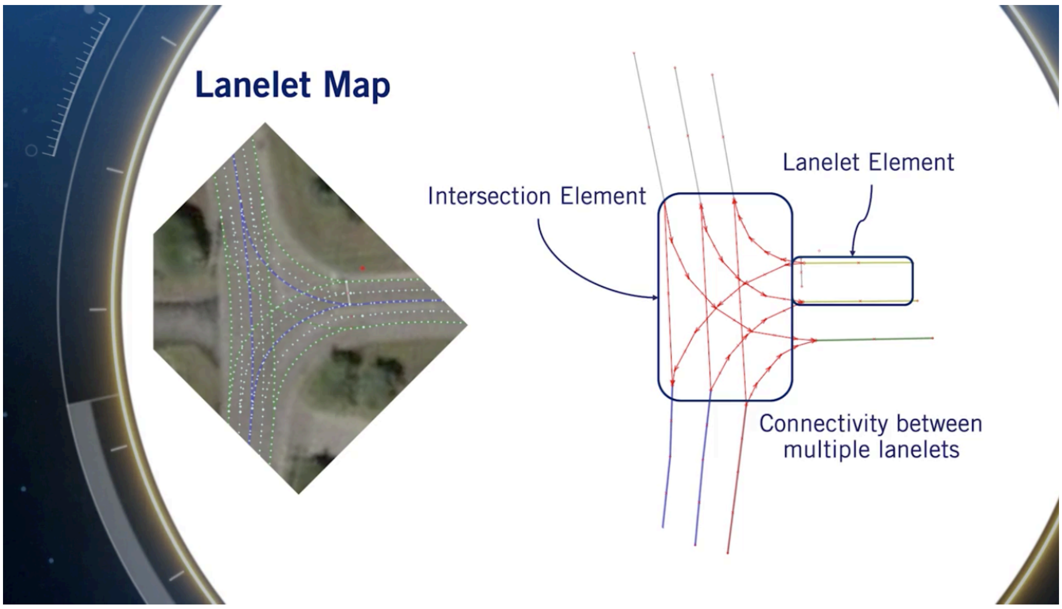
- Perception stack which must detect and track all dynamic objects in the scene.
- The 3D bounding box of the detected dynamic object is used to remove all the points in the affected area.
- Perceptionの方法はまだunsatisfactory。
  - Not all vehicles are dynamic, so they should be included (in occupancy grid).
    - 解決：History of dynamic object location can be used to identify parked vehicle.
    - つまりtrackingを利用。vehicleが止まっているかをチェック。
  - The dynamic objects are identified from the previous LIDAR frame.
    - 解決：Predicted future location improvement.
    - Rely on predictions of the moving objects' motion based on their object tracks.
    - The bounding box is shifted forward along the predicted path, leading to a greater amount of the points of the dynamic object being removed from the most recent lidar scan.
    - removal処理のデータはいつも最新のlidar scan、dynamic objectの予測は以前のlidar scanベース。

Projection of LIDAR Onto a 2D Plane
- Simple solution:
  - Collapse all points by Zeroing the Z coordinate.
  - Sum up the number of LIDAR points in each grid location.
    - More points indicated greater chance of occupation of that grid cell.

Lesson 4: High Definition Road Maps

内容
- Lanelet map.
  - OpenDriveとどう違うんだ！
- The elements that make up a lanelet map.
  - Lanelet element.
  - Intersection element.
  - Operations that can be done on lanelets.
- Creation of lanelet maps.
- Connectivity between lanelets.

High Detailed Road Map
- While a traditional map stores the approximate locations of roads, the High Definition Road Maps stores the precise road locations including all lanes down to a few centimeters accuracy.
- また、the High Definition Road Maps store all of the locations of road signs and signals.



Lanelet Map
- 2014年の論文。OpenDriveよりは随分新しい!
- Lanelet2: https://github.com/fzi-forschungszentrum-informatik/Lanelet2
- Due to this method's effectiveness in storage and communication of the complex set of information needed for HD mapping, it is widely used.
  - なのに、初めて聞いた!
- This will play a key role in the behavior planning methods in particular.
- Intersection Element.
  - for simple retrieval during motion planning tasks.

Lanelet Element
- Defines the following.
  - Left and Right boundaries.
    - boundaryって道路の? Laneの?
      - Laneのです: https://github.com/fzi-forschungszentrum-informatik/liblanelet
      - Boundaries of the lanelet are represented as the edges of the lanelet.
        - lane markingsもしくはcurbs.
  - Regulation.
    - that might be present at the end of the lanelet element, such as a stop sign line or a static sign line.
      - Note that we only store a line for any regulatory element as this is the point that the autonomous vehicle treats as the active location for that regulatory element.
      - endってどういう意味? endだけ?
        - なぜならLaneletの区切りはそう設計している!
    - Elements.
      - Which come at the end of the lanelet.
      - Represented as lines which are defined by a set of co-linear points.
      - Regulation elements usually require an action or decision to be made.
      - 例えば、Stop line, Traffic lights line, Pedestrian crossings.
    - Attributes.
      - Which affect the entirety of the lanelet.
      - such as speed limit.
      - such as whether this lanelet crosses another lanelet as in an intersection or merge.
  - Connectivity to other lanelets.
- Each lanelet ends as a regulatory element or a change to a regulatory attribute is encountered.
  - つまりspeed limitが変わるところ、またはstop signがあるところは、laneletの区切りです。
  - This means that a lanelet element can be as short as only a few meters in the case of a lanelet, which is part of an intersection, or can be hundreds of meters long for a highway road segment.
- Lane boundaries are stored as a set of points creating a continuous polygonal line.
  - The distance between points can be as fine as a few centimeters, or as coarse as a few meters depending on the smoothness of the polyline in question.
    - 間隔は一定じゃなくてもいい。
  - The ordering of the points defines the direction of travel and heading for the lanelet.
    - 白線点列の順番はここも大事ですね!
  - A center line between the two boundaries can be interpolated, which can be used as the desired path of travel for the autonomous vehicle in that lane.
    - Basepath点列は特に要らない。大事なのは両側境界線点列。しかし、2車線の場合、重複の点列を保存するようですね。でもこの方が綺麗で扱うのも簡単です。
- Each lanelet has 4 possible connections: 左右前後。
- The entire lanelet structure is connected in a directed graph, which is the base structure of the HD map.

Operations Done On Lanelets
- Path planning through complex road networks.

- Localize Dynamic Objects.
- Interactions with other Dynamic Objects.

Creations Of Lanelets
- コース１に既にあった：offline, online, offline creation with online updating.

宿題Programming Assessmentのお陰で、occupancy grid mapsを生成する4つステップに直感ができた！

1. get_ranges()
    1. Gather range measurements of a moving car's surroundings using a lidar scanning function.
    2. これはlidarのmeasurementsを模擬している。結果はlidarの各方向（bearing）のrangeを返す。obstacleがあったら、obstacleまでの距離。obstacleがなかったら、max rangeを返す。
2. inverse_scanner()
    1. Extract occupancy information from the range measurements using an inverse scanner model.
    2. lidarのranges情報からoccupancy grid mapを作る。各grid cellにno information, high probability, low probabilityの3種類probabilityをつける。
3. Perform log odds updates on an occupancy grids based on incoming measurements.
    1. log oddsで更新が便利なので。
4. Iteratively construct a probabilistic occupancy grid from those log odds updates.
    1. log oddsからprobabilityに戻す。