Automatic Threat Recognition at LLNL

T04 Program Review, Boston, MA

Nov. 6, 2014

Philip Top, Ana Paula Sales, Hyojin Kim, Eric Wang,
Jay Thiagarajan, Timo Bremer, Steve Azevedo, Harry Martz

Lawrence Livermore National Laboratory

LLNL-PRES-663046

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344. Lawrence Livermore National Security, LLC

LLNL team

- Harry Martz (Program Lead)
- Steve Azevedo (Project Management)
- TO4 team funded through DHS-COE directly
  - Philip Top (Electrical Engineer)
  - Ana Paula Sales (Classification and Statistics)
- Leveraged by LLNL internally-funded R&D
  - Timo Bremer, Eric Wang, Hyojin Kim, Jay Thiagarajan

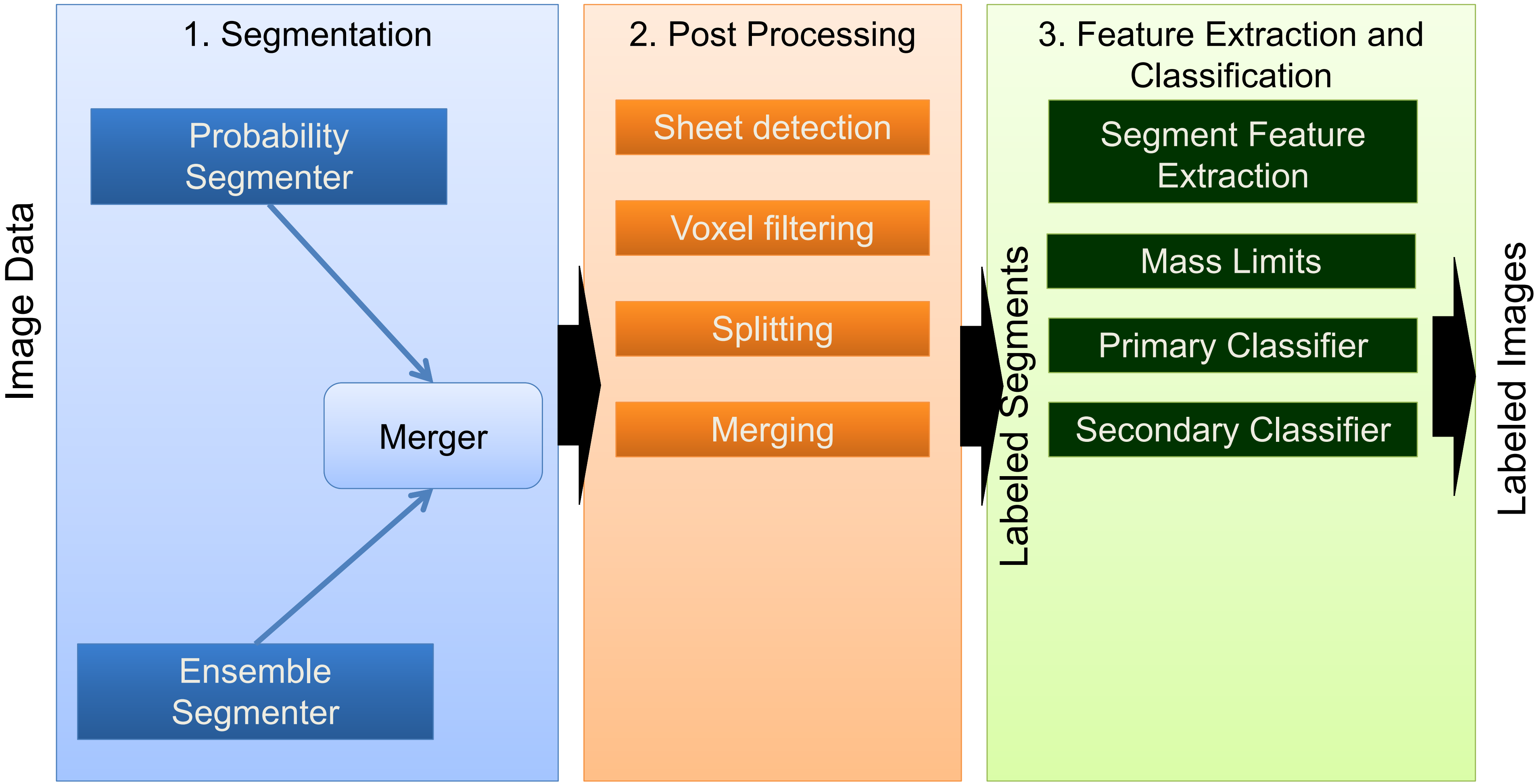
Summary of P_D/P_{FA} Results

				No special rules (except for PT sheets)	
Target Type	Target Subtype	Level of Difficulty	Num Targets	Num Detected	PD [%]
Target	All	All	407	381	**93.6**
Target	Clay	All	111	107	96.4
Target	Rubber	All	158	150	94.9
Target	Saline	All	138	124	89.9
Target	Bulk	All	270	251	93
Target	Sheet	All	137	130	94.9
Target	All	Low	77	75	97.4
Target	Clay	Low	29	29	100
Target	Rubber	Low	22	22	100
Target	Saline	Low	26	24	92.3
Target	Bulk	Low	56	54	96.4
Target	Sheet	Low	21	21	100
Target	All	High	317	294	92.7
Target	Clay	High	82	78	95.1
Target	Rubber	High	125	118	94.4
Target	Saline	High	110	98	89.1
Target	Bulk	High	201	185	92
Target	Sheet	High	116	109	94
Pseudo-target	Sheet	High	10	10	**100**
			Num Non-targets	Num FAs	PFA [%]
			1371	163	**11.9**
				Num Scans with FAs	Avg Num FAs
				110	1.57

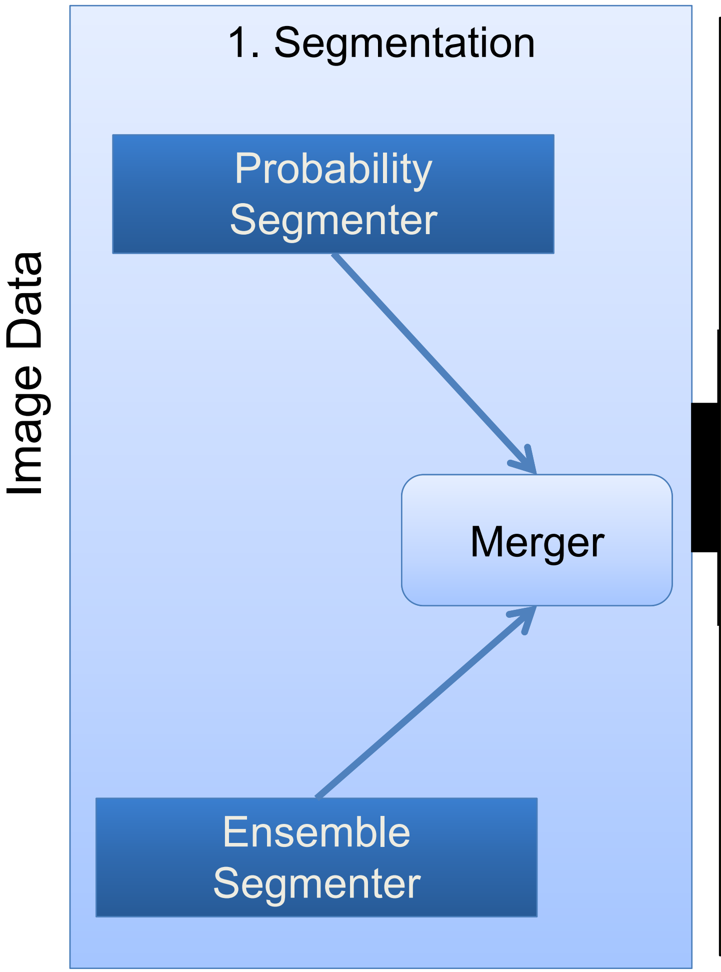
No special rules:
93.6% / 11.9%

(special post-processing for pseudo-target sheets)

ATR Pipeline



ATR Pipeline



What is novel?

- Two Segmenters in parallel, then merged
- Probability Segmenter
  - Use of voxel slabs to generate features for the voxels
  - Concentrate efforts on medium- to high-probability target voxels
  - Uses a random forest algorithm
- Ensemble Segmenter
  - Based on a bottom-up hierarchical segmentation
  - Creates an ensemble of hierarchical segmentations by randomizing the merging order
  - Combines high-level object semantics with low-level local features into the hierarchy
  - Final object segmentation using graph cuts

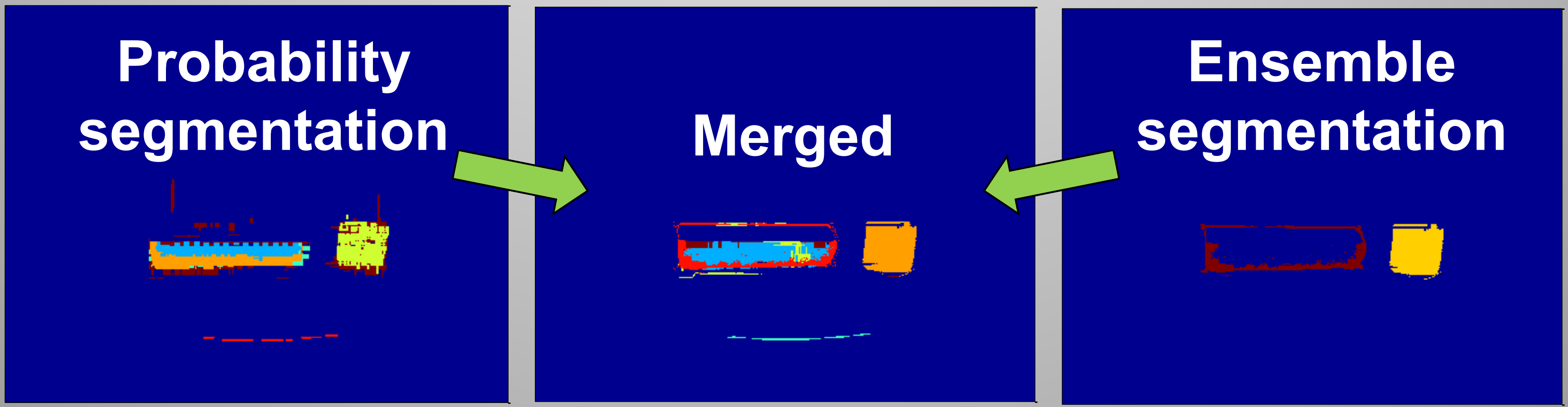
The motivation was to provide a baseline of ATR performance

- Develop an ATR Pipeline that is
  - Compatible with new targets
  - Separable – Each component of the pipeline can be evaluated independently and as part of the whole
- Allow selection of algorithms and design parameters
  1. Segmentation – Each segmenter has drawbacks and advantages, so we merge two types
  2. Post-processing – After merging segmentation results, apply additional information such as sheet separation, artifact reduction
  3. Classification – Employ multi-stage feature extraction and classification

1. Segmentation

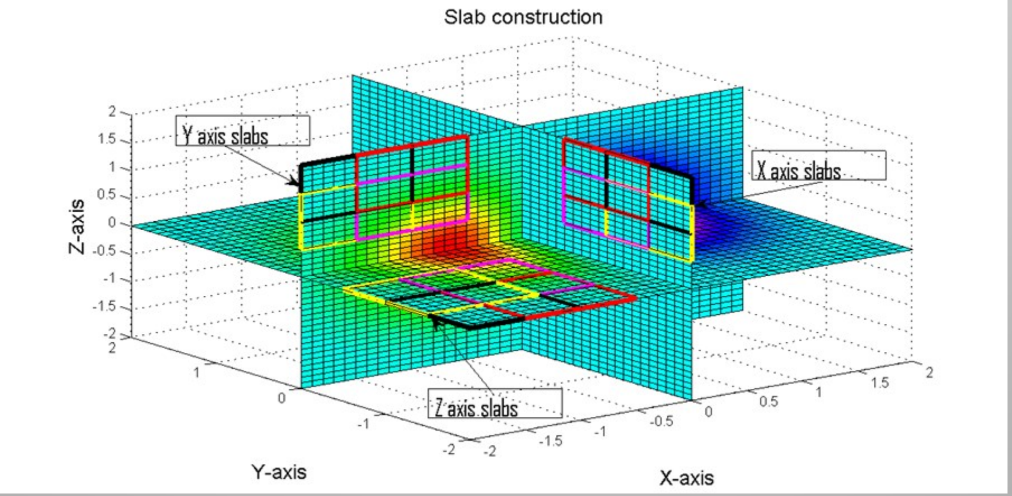
- **Probability segmenter**
  - Compute a probability that each voxel belongs to a target and merge connected voxels together
  - Goal of 100% recall
  - Tends to merge targets together
  - Poor precision

- **Ensemble segmenter**
  - Generate an ensemble of potential segments and compute based on average behavior
  - Good segmentation
  - Goal of >90% detection
  - It misses several sheet objects



Technical Description of Algorithm: Probability Segmenter (1 of 2)

- Break image into 10x10 voxel “slabs” in each plane X,Y,Z (planes, not cubes)
- Generate a feature vector from the slab
  - median, stdev, range, type-dependent features based on the discrete cosine transform for texture
- Compute the probability the slab belongs to a target of interest (clay, saline, rubber, and powder), or below a threshold

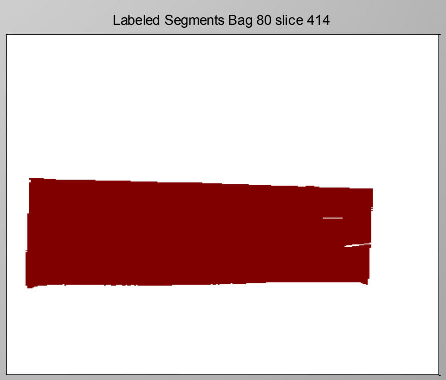


Technical Description of Algorithm: Probability Segmenter (2 of 2)

- Subject the identified slabs to 3D connected-component labeling
  - Only slabs that are connected (via adjacency) to at least K other slabs (to form segments of a minimum size) are retained
  - Slabs that are not connected to enough other slabs are discarded
- The output of the Probability Segmenter are these “rough segments”

Bag 80 y-slice 414

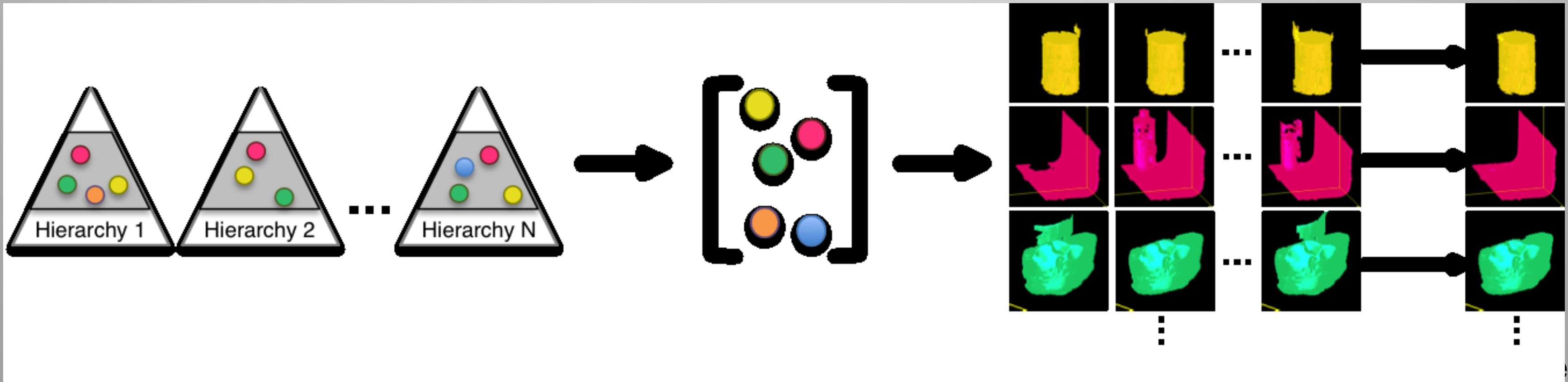
Compute probabilities



Threshold and find connected voxels

Technical Description of Algorithm: Ensemble Segmenter (1 of 2)

- Creates an ensemble of hierarchical segmentations by randomizing the merging order of local features (attenuation, histogram)
- Include high-level object semantics (e.g., surface/volume ratio) with low-level local features into hierarchy of candidate objects
- Combine localized candidate objects into final objects using consensus segmentation with graphcuts

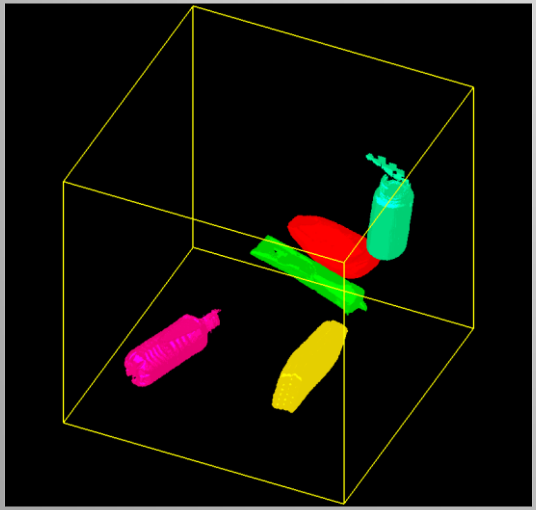


…ects

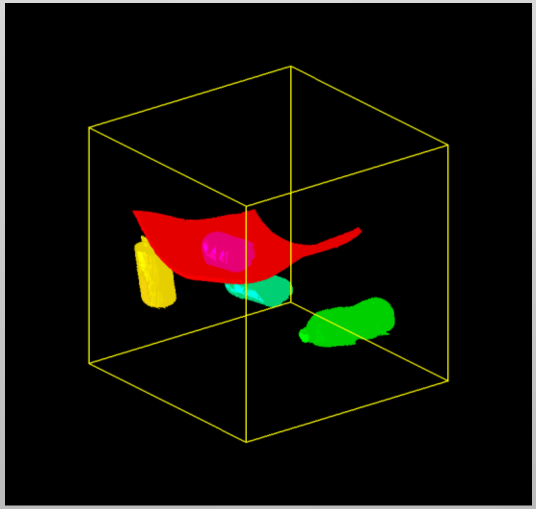
segmentations objects objects

Technical Description of Algorithm: Ensemble Segmenter (2 of 2)

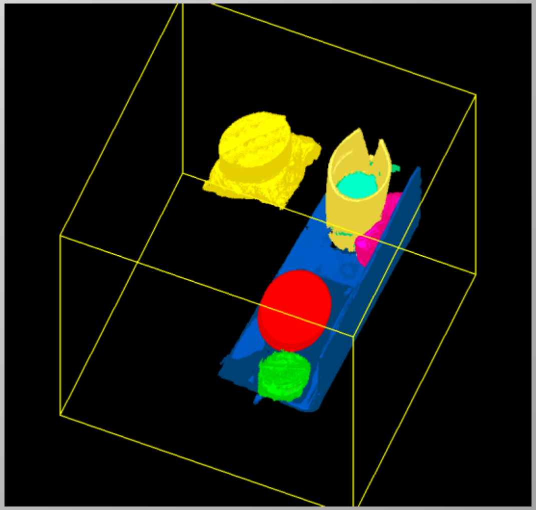
- For many object types, it compensates for reconstruction artifacts
- Objects can be identified from a wide range of levels in the hierarchy
- Can be customized for how much segmentation is desired
- Converges to the “average” behavior with consensus segmentation



SSN: 088



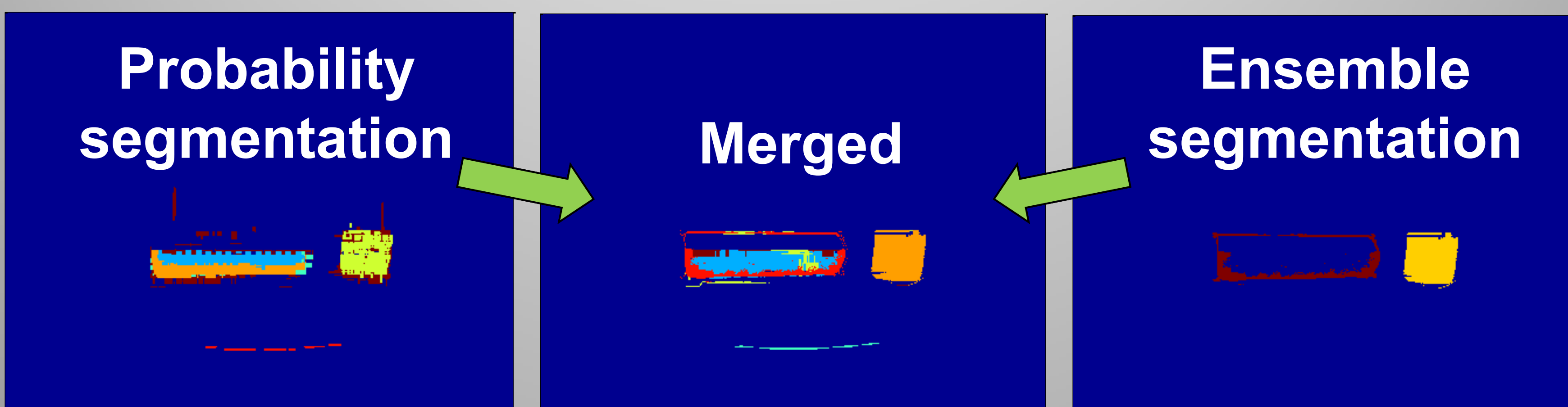
SSN: 093



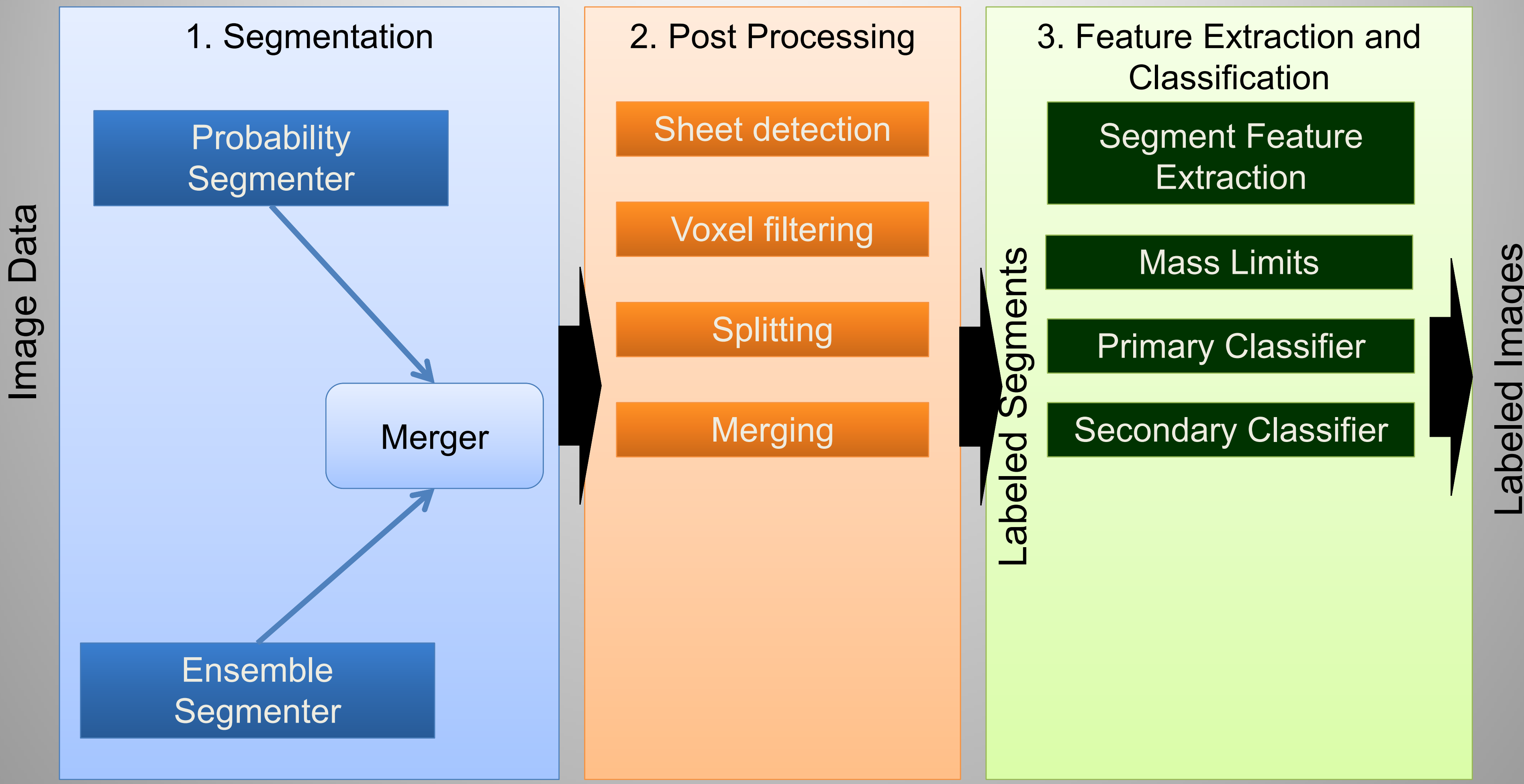
SSN: 094

1. Flexible Segmentation Merger

- Use segments from Ensemble Segmenter unless very few pixels are found in the Probability Segmenter (# pixels is flexible)
- Make new segments out of the remaining Probability Segmenter voxels
- A segment can be split if only half is in Probability Segmenter
- Complexity can be traded off with the post-processing



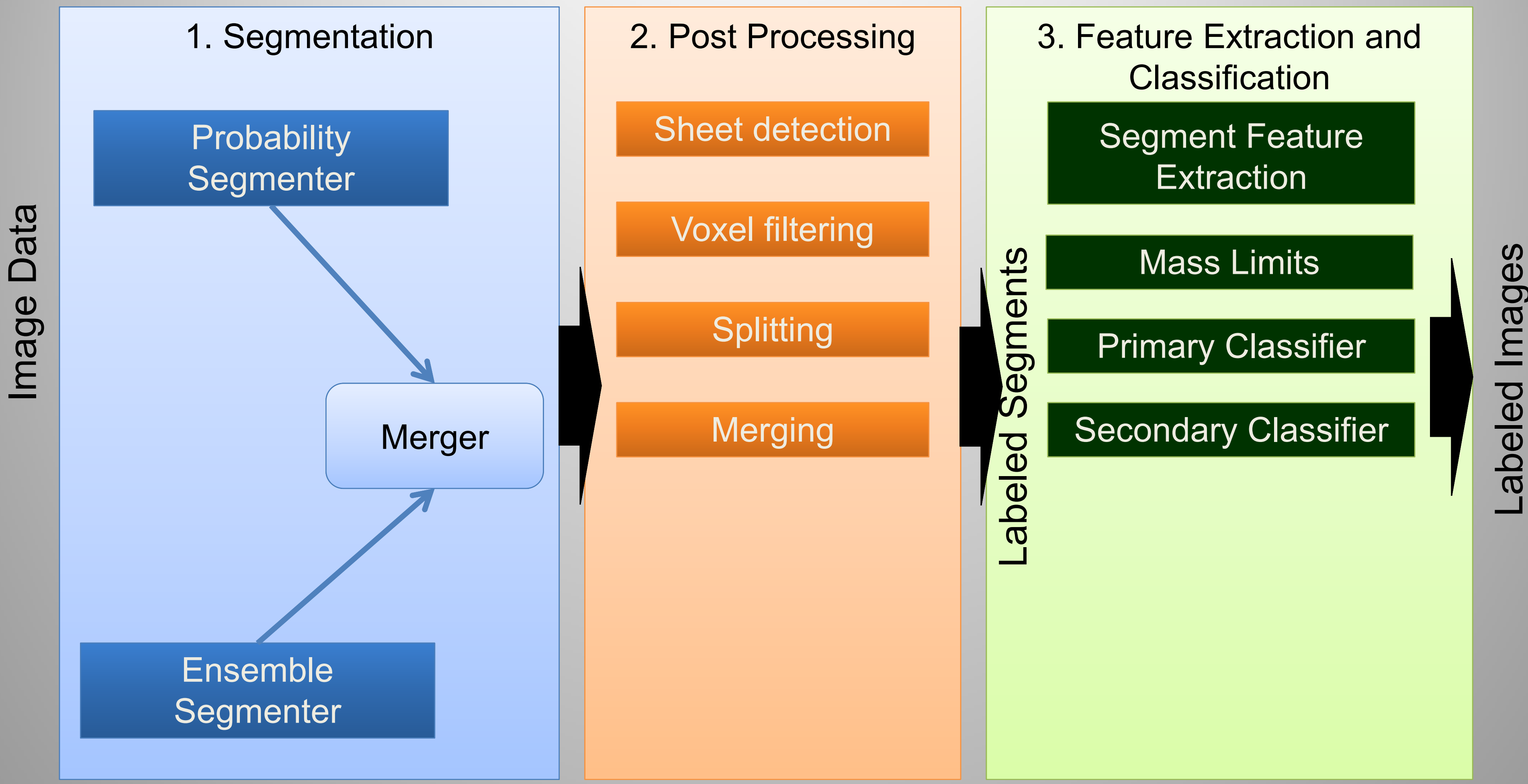
ATR Pipeline



2. Post-processing

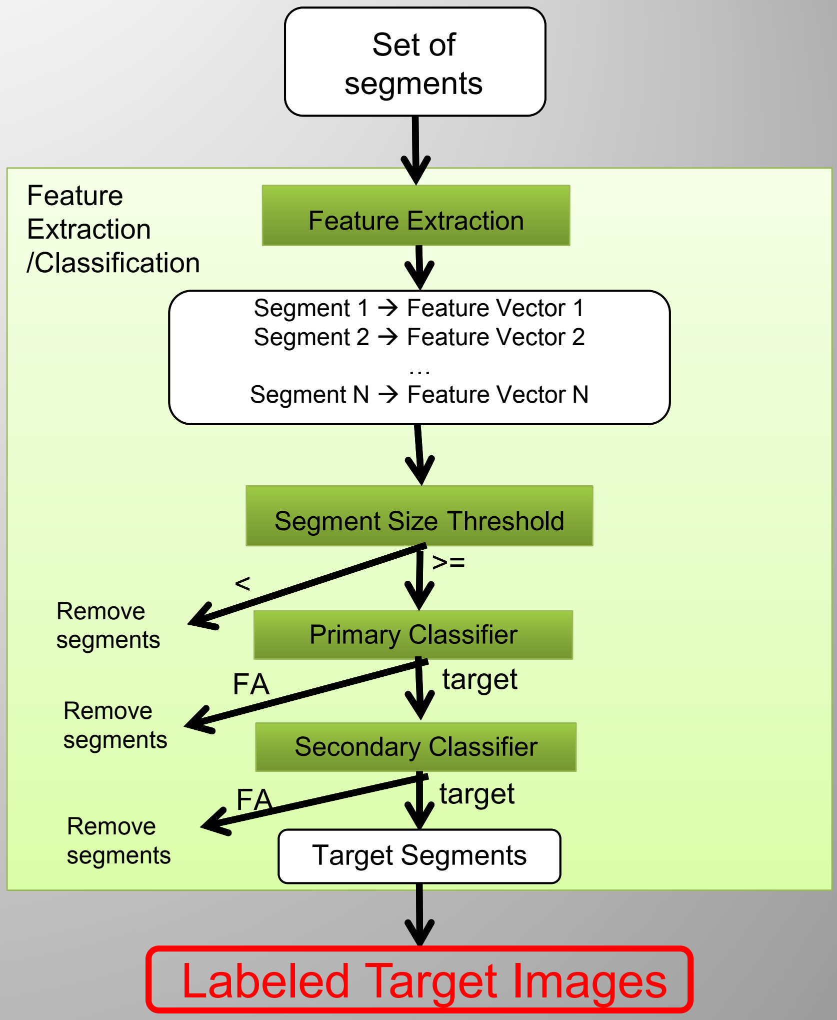
- Performs further splits of segments
  - Are large sections of segments only connected by a narrow channel or not connected? (If so, split.) OR
  - Are there multiple statistically separable histogram peaks? (If so, split.) OR
  - Can one of the adjoining segments be characterized as a sheet? (If so, remove any large clusters.)
- Performs further merges of segments
  - Are segments close together or overlapping? AND
  - Do they have the same statistical properties? AND
  - Were they previously separated? AND
  - Do they fit together?
  - (If so, merge them.)

ATR Pipeline



3. Feature Extractor + Classifier (1 of 2)

- Extract features for each segment in the labeled image
  - Voxel slabs (10x10) in the X, Y, and Z axis are computed as described earlier
  - Features include:
    - Mean of all pixels in segment
    - Voxel slab mean
    - Voxel slab standard deviation
    - Pixel count
- Feature vectors are fed into a classifier (settled on Random Forest)
  - Primary: uses no special rules
  - Secondary: some customization
- Segments called “false alarms” by the classifier are removed from the final labeled targets image



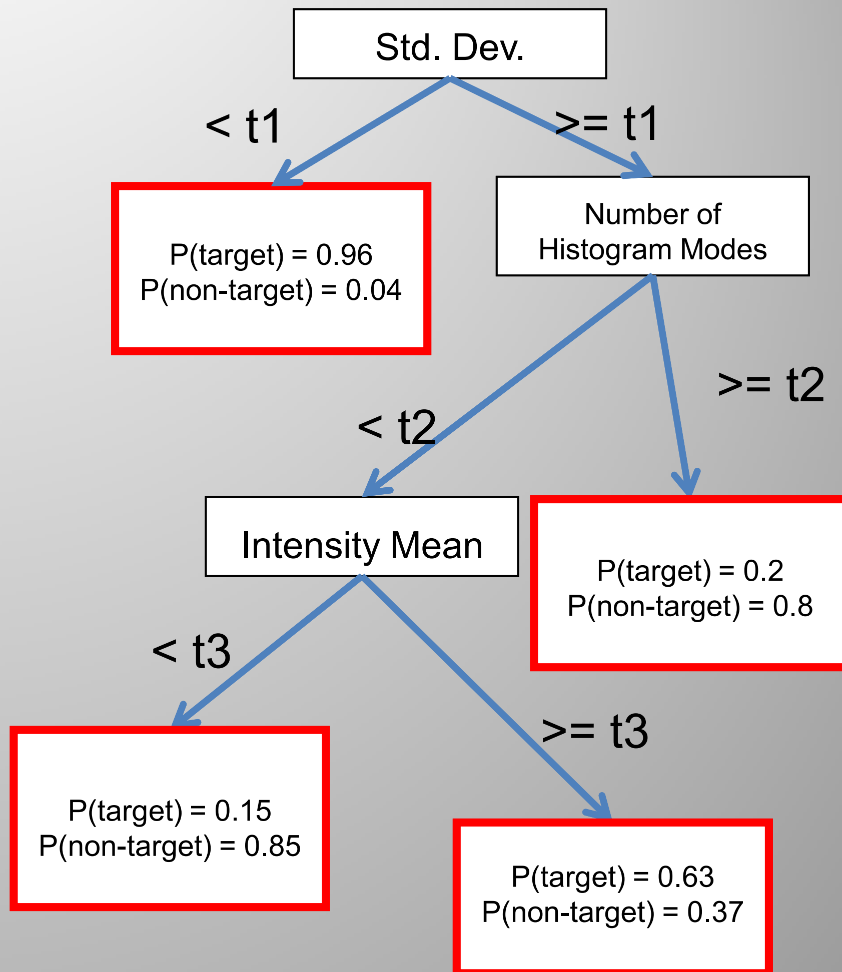
3. Feature Extractor + Classifier (2 of 2)

- Feature Extractor
  - Segment size threshold: segments smaller than a certain threshold are removed (i.e., labeled as non-target)
  - Pseudo-target sheet threshold: segments that are within a certain range for a number of features are retained in the final labeled images
    - Examples of features are mean, mode, and standard deviation of attenuation, number of peaks, cosine transform
- Primary Classifier – Operates on the entire set
  - Used a Random Forest (RF) algorithm
  - We use 3 RFs: train on 1/3 of the data, and evaluate the other 2/3; for each 1/3 of data
- Secondary Classifier – Allows other rules to reduce FA
  - Provides further filtering for the segments that pass the Primary Classifier
  - Particular rules for pseudo-target sheets
  - Also based on RFs, but training and evaluation sets differ somewhat

Random Forests Classifiers

- A Random Forest is an ensemble of decision trees
  - Features are selected at features
- Decision trees
  - Provide a partitioning of the feature space of the data into disjoint sets
  - Each partitioning is associated with a probability vector of the possible outcome classes
  - Classification of a new object is done by mapping features to the partitioning of the data
  - The label is defined by the probability vector

Artificial Example of Decision Trees



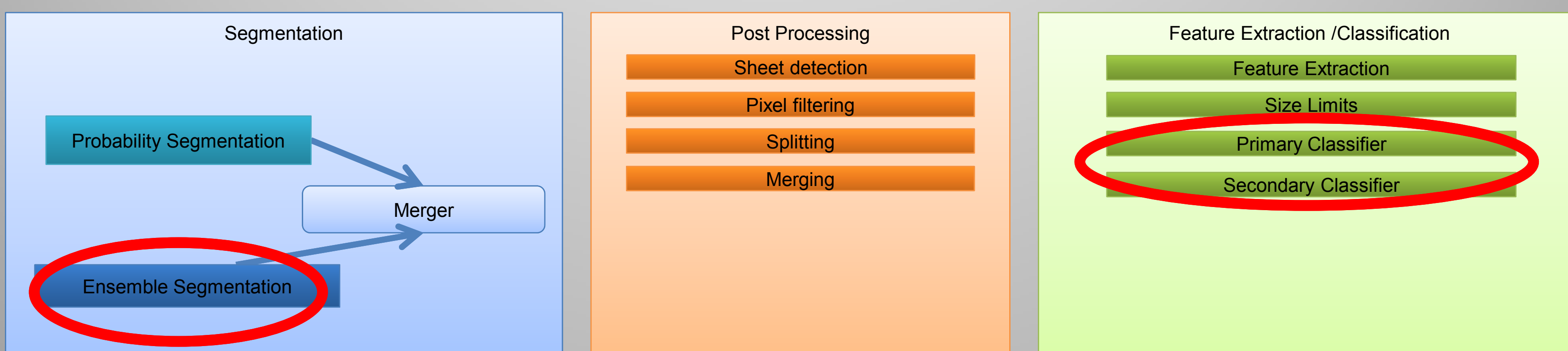
Other Classifiers were explored

(that did not work as well as Random Forests)

- Adaptive Boosting
  - An ensemble classifier where the outputs of classifiers are weighted according to how weak/strong they are. Weak classifiers are tweaked in favor of those instances misclassified by previous classifiers
- Artificial Neural Networks
  - Algorithm inspired by how information is transmitted in the brain via neurons. Large number of inputs are approximated by layers of neurons whose connections are learned.
- Naïve Bayes
  - Probabilistic classifier based on Bayes theorem. It assumes independence of features.
- Nearest Neighbors
  - Provides simple data interpolation in one or many dimensions. It clusters the training data, and each cluster is represented by its centroid. New observations are assigned to the cluster whose centroid is most similar to itself.
- Support Vector Machines
  - Obtains data classification by identifying an optimal hyperplane that separates the two classes under consideration.

ATR Training

- How was over-training on supplied data prevented?
  - Majority of steps are unsupervised – no special rules
  - Supervised steps use three-fold cross-validation (1/3 training, 2/3 evaluation)
    - — Use multiple classifiers such that training data never overlaps with evaluated data



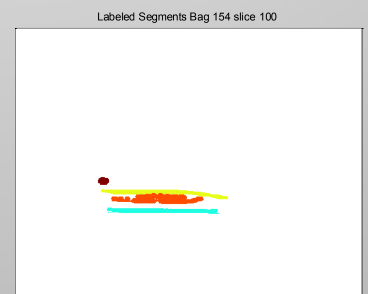
Includes supervised information

ATR Training

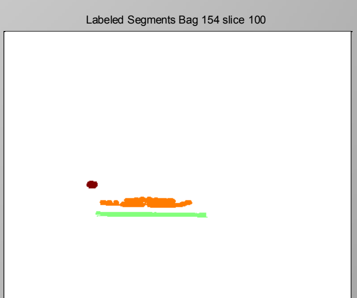
- How were false alarms reduced?
  - By the use of multiple staged steps
    - — Probability Segmenter: Reduces the number of voxels used in segmentation. It is tuned to have nearly 100% recall and minimize the number of false alarms.
    - — Classifier: Labels the segments as “targets” or “false alarms”, such that only the “target” segments are included in the final set of labeled images.

CT

Ground Truth



After Preprocessing



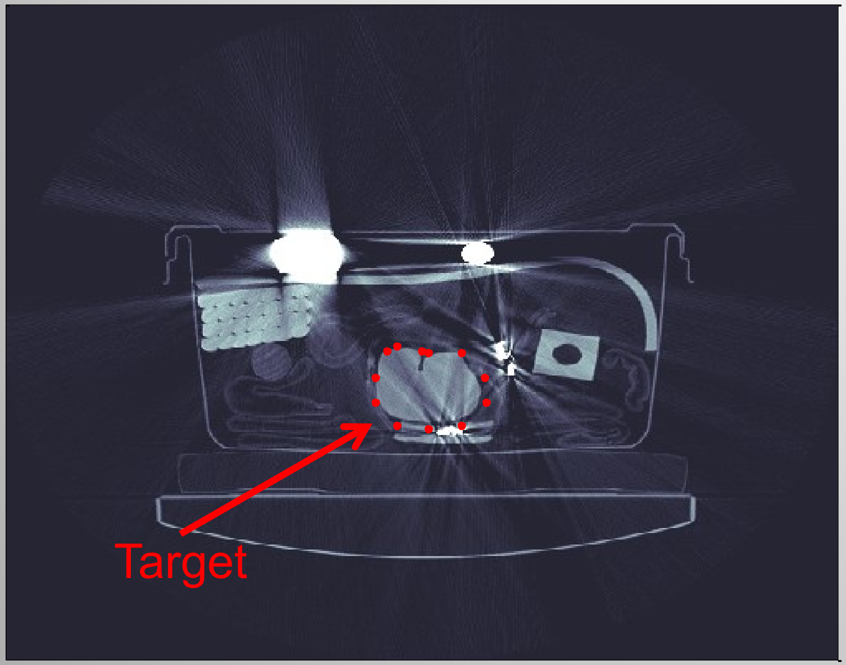
Final

Robustness to new targets

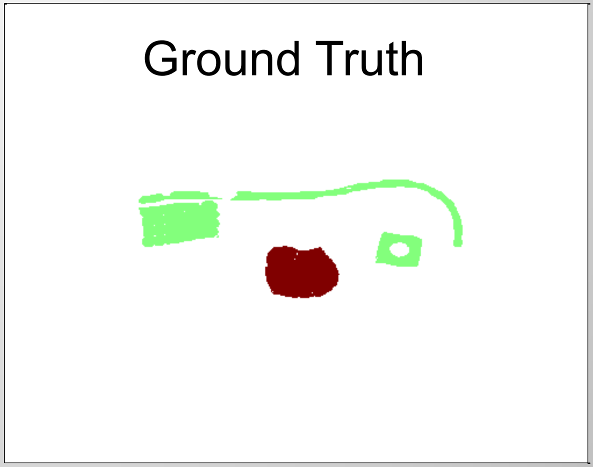
- Each of the target types has a separate path in the pipeline… starting from segmentation
- This facilitates the addition or removal of targets
- We can use simulated data to detect new target classes; all that is needed are the features

Case #1: Bulk with bad streaks caused by metal

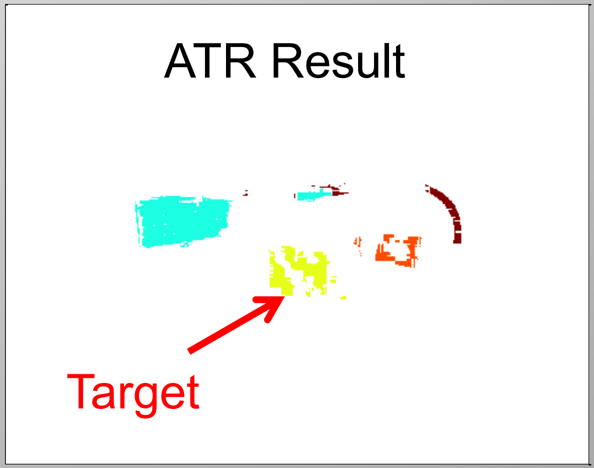
Bag 13 slice 105



Ground truth Bag 13 slice 105



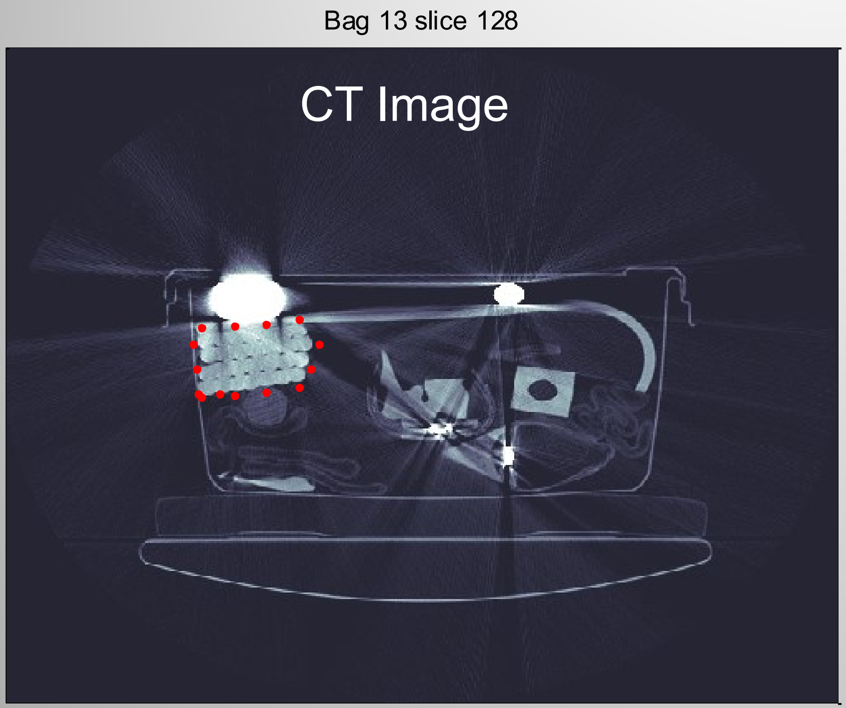
Labeled Segments Bag 13 slice 105

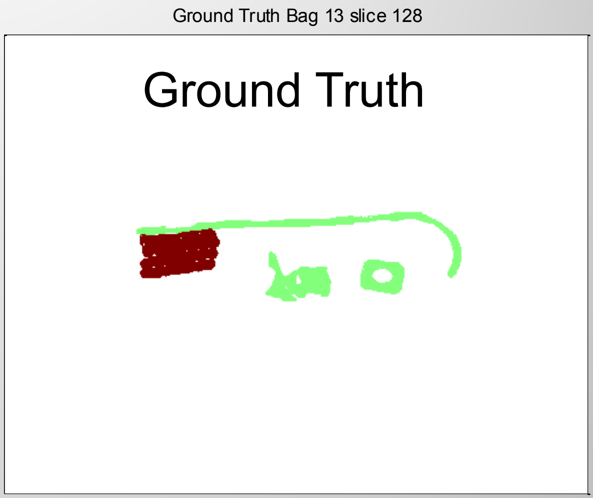


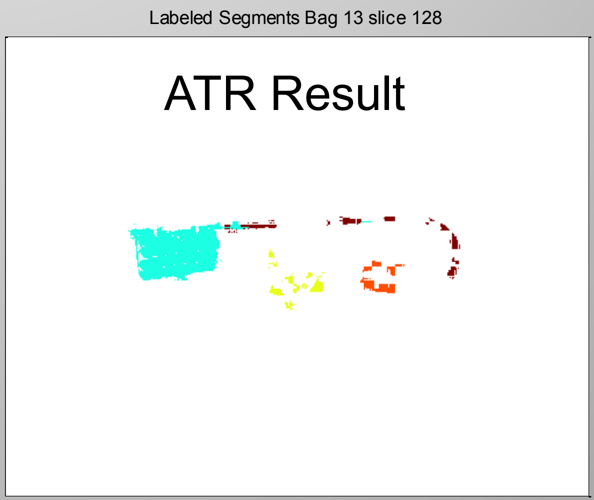
Detected: Yes
Precision: 95.2% recall: 60.1%

Streaks cause difficulty to the final classifier stage

Case #2: Bulk with bad shading caused by beam hardening and scatter





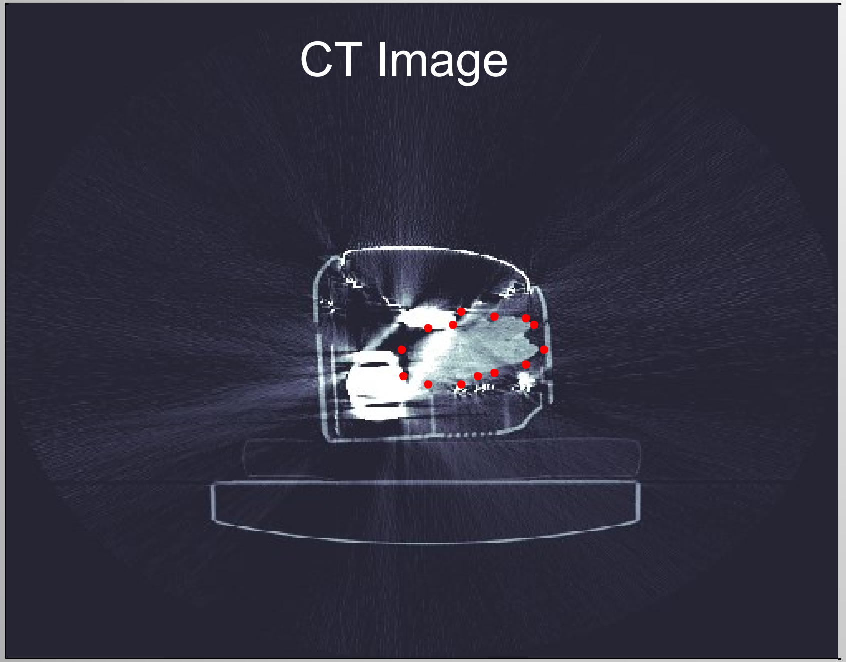


Detected: Yes
Precision: 72.2% recall: 94.2%

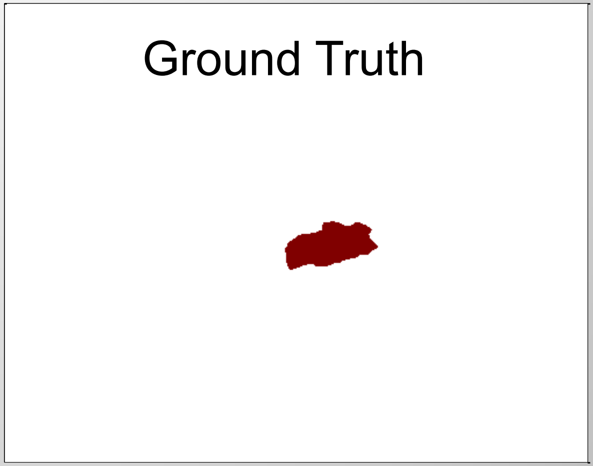
Partially merged with nearby sheet

Case #3: Bulk inside electronics

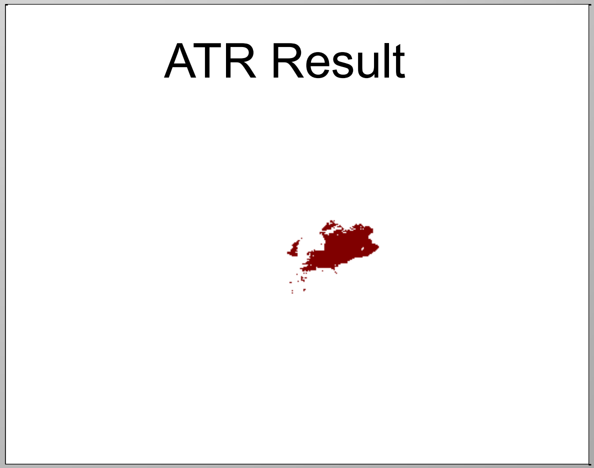
Bag 35 slice 49



Ground Truth Bag 35 slice 49



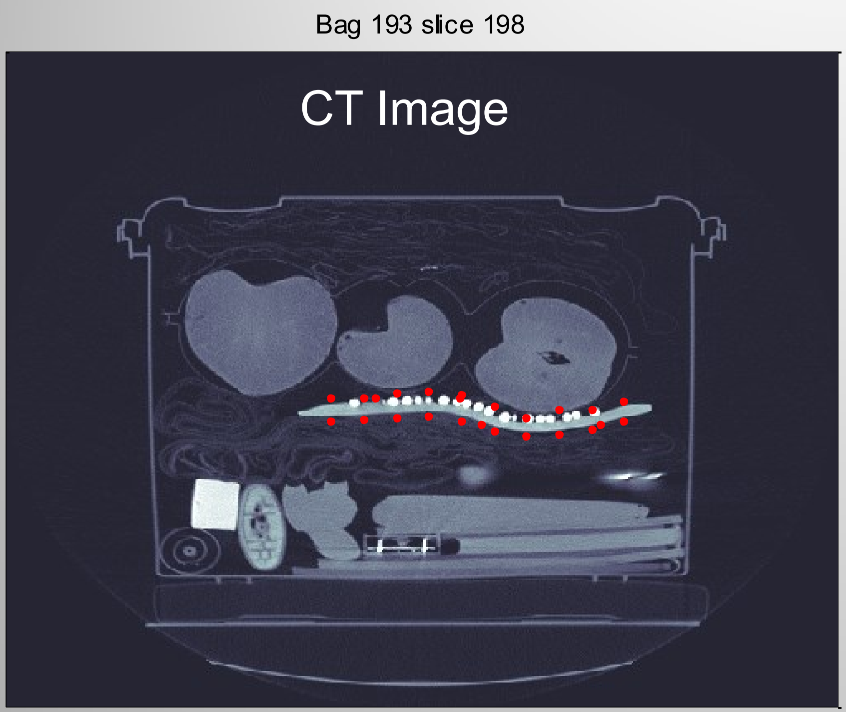
Labeled Segments Bag 35 slice 49

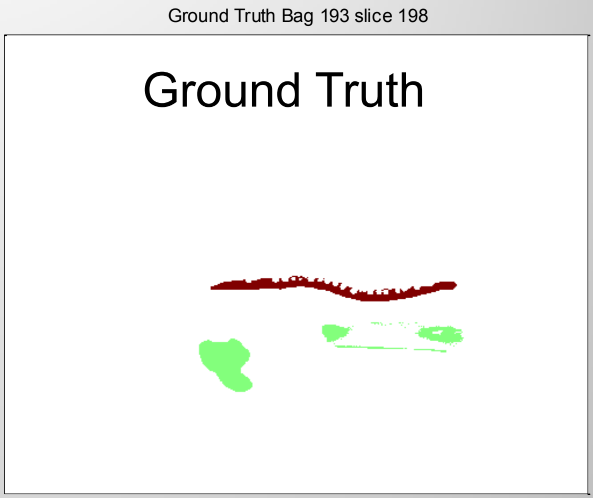


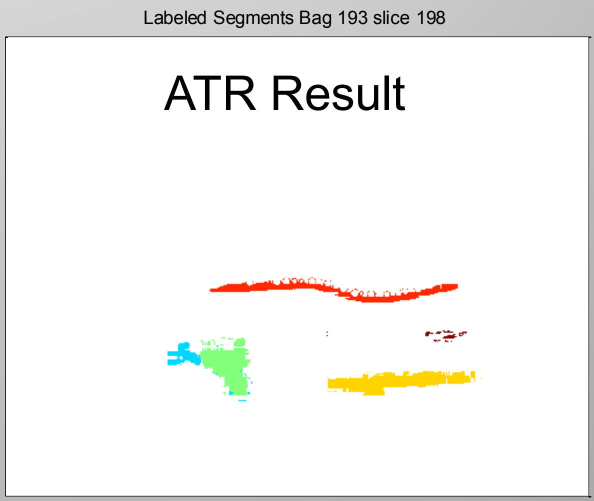
Detected: Yes
Precision: 87.5% recall: 79.3%

Not fully captured

Case #4: Bulk with texture





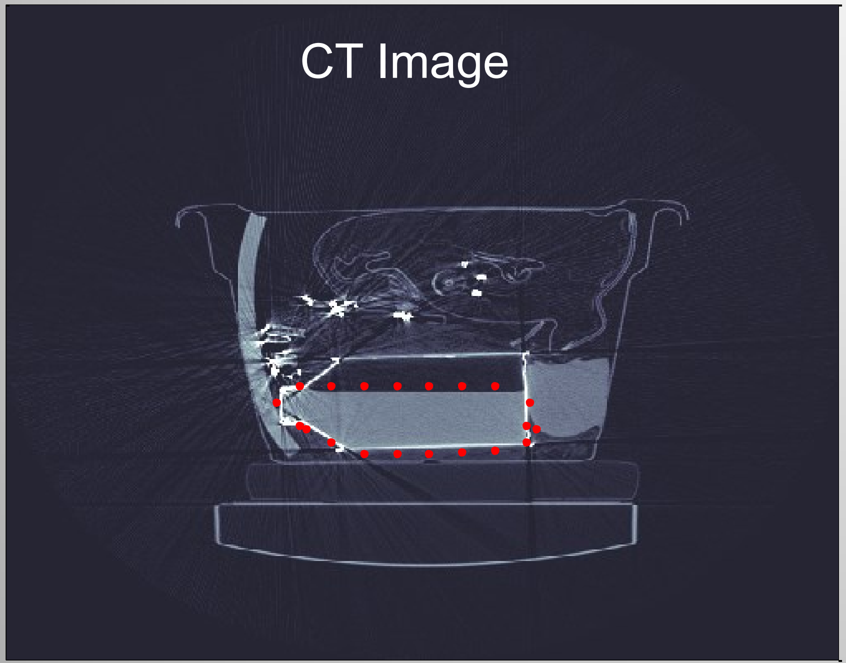


Detected: Yes

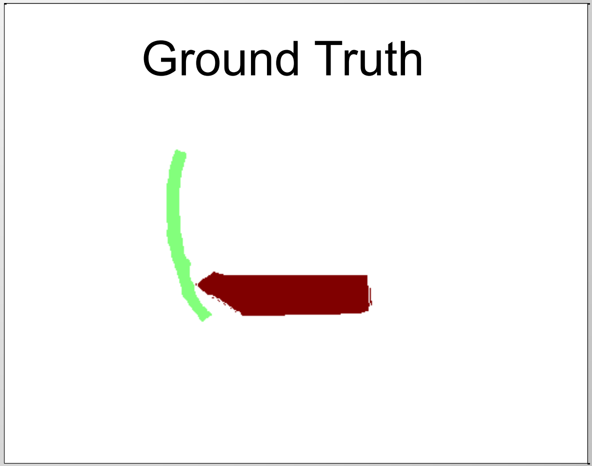
Precision: 96.5% recall: 73.8%

Case #5: Bulk with density close to water (~5% saline)

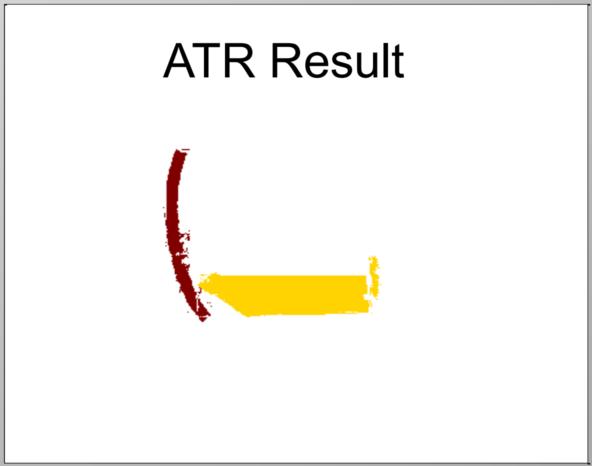
Bag 63 slice 45



Ground Truth Bag 63 slice 45



Labeled Segments Bag 63 slice 45

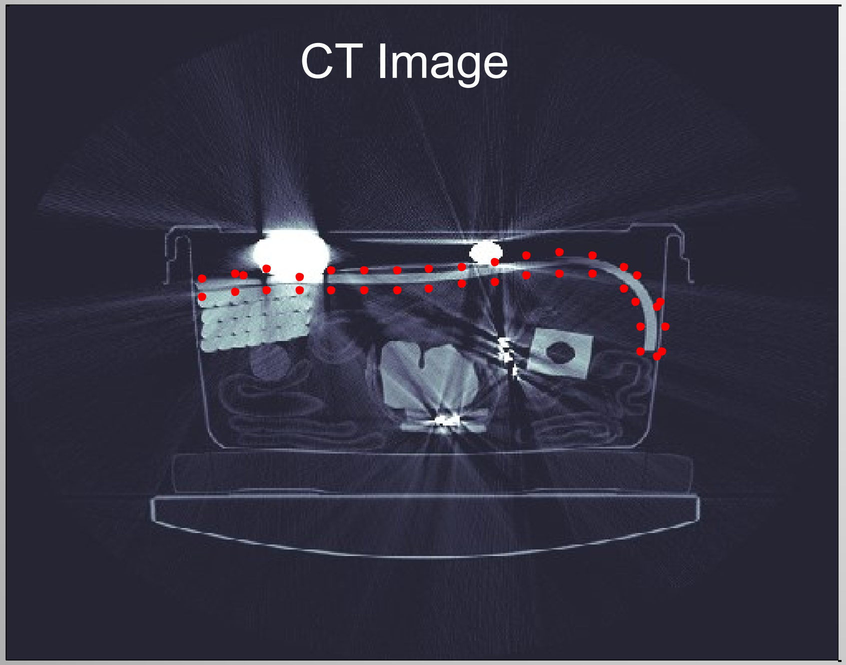


Detected: Yes

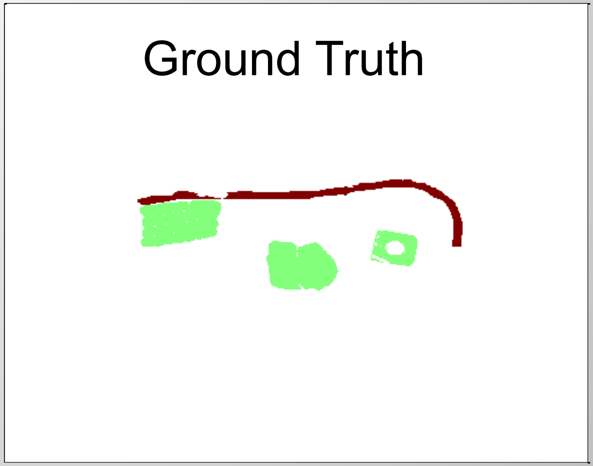
Precision: 93.0% recall: 95.5%

Case #6: Sheet with bad streaks caused by metal, beam hardening and scatter

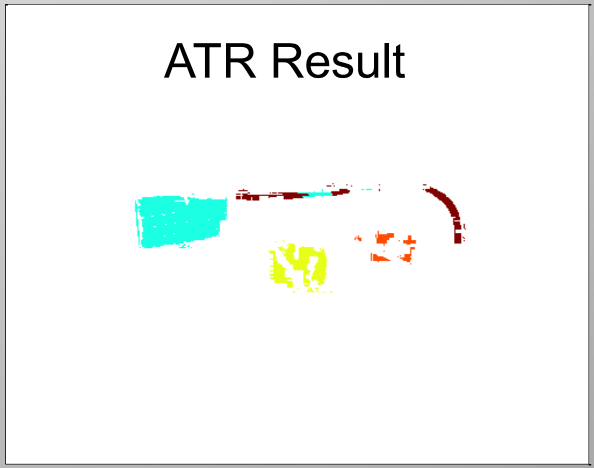
Bag 13 slice 111



Ground Truth Bag 13 slice 111



Labeled Segments Bag 13 slice 111

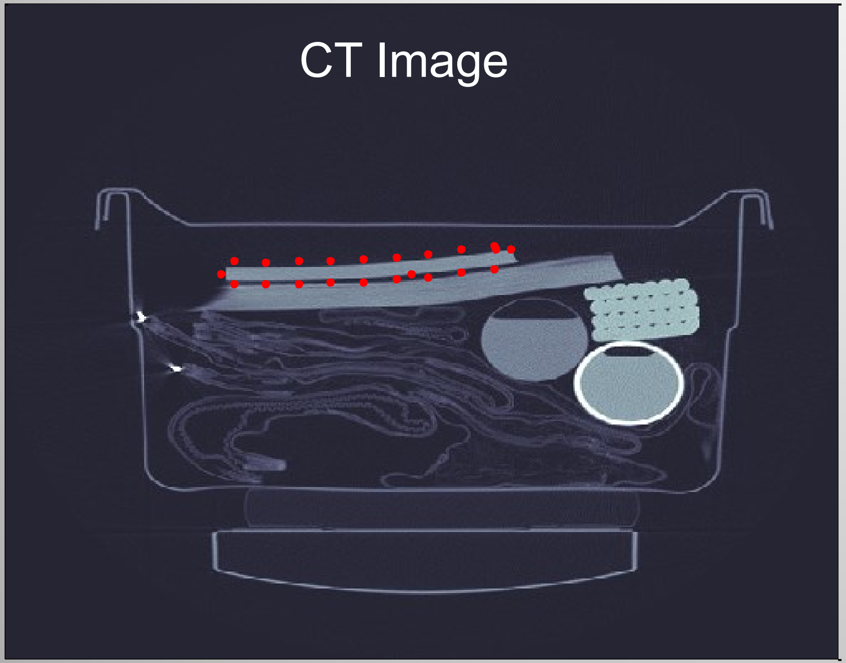


Detected: Yes
Precision: 83.3% recall: 26.7%

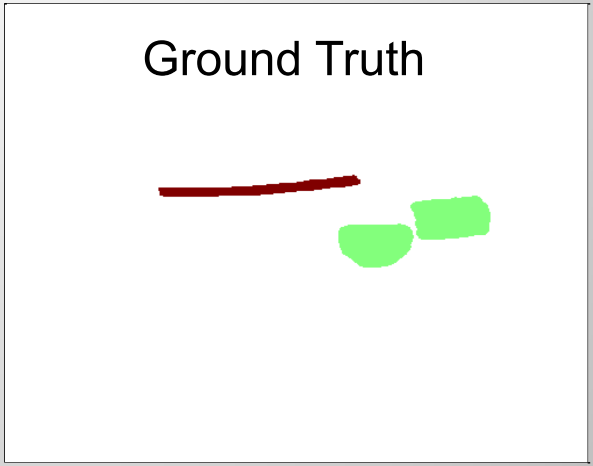
Split into a couple pieces;
Not fully captured

Case #7: Sheet laying on top of another flat object

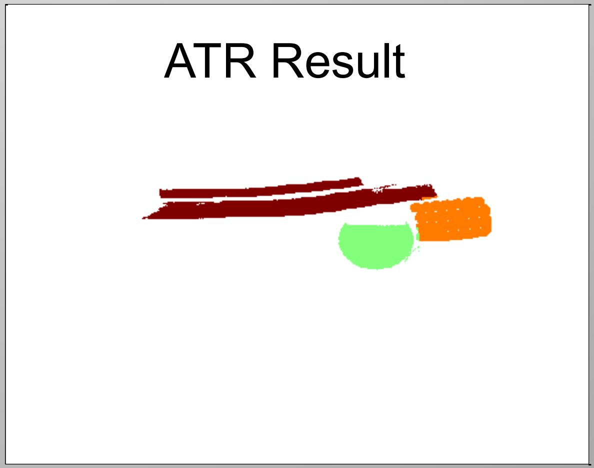
Bag 33 slice 46



Ground Truth Bag 33 slice 46



Labeled Segments Bag 33 slice 46

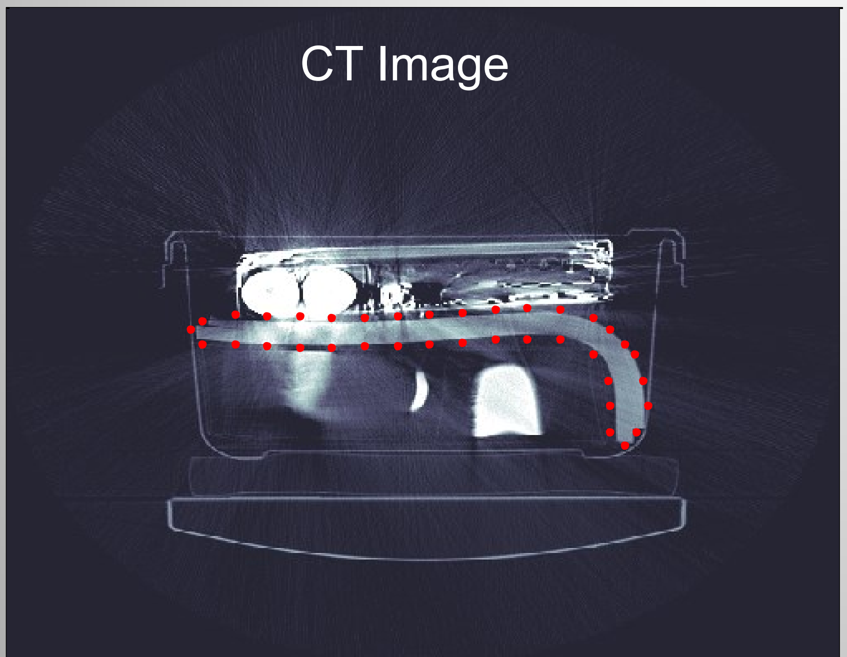


Detected: Yes
Precision: 21.1% recall: 82.7%

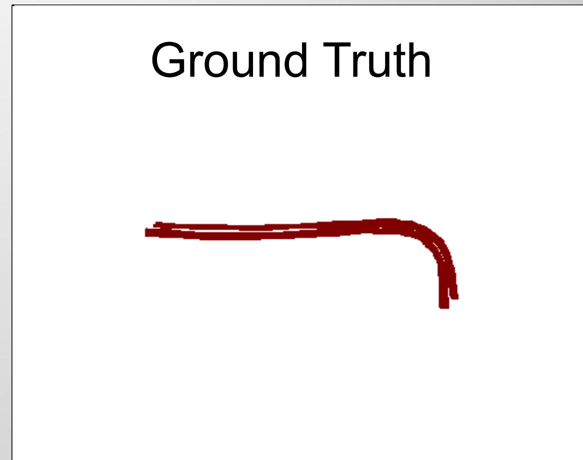
Merged with object below it

Case #8: Object with lots of photon starvation

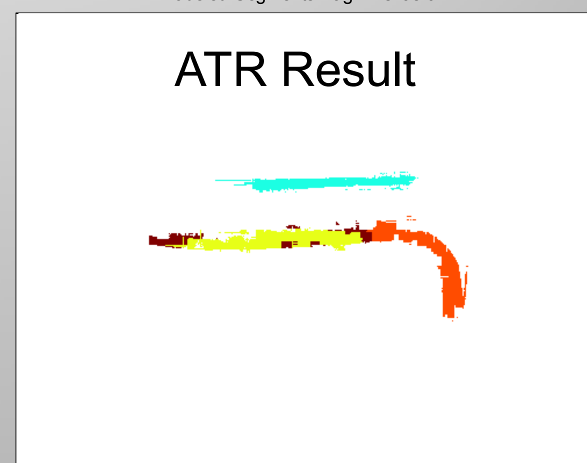
Bag 11 slice 94



Ground Truth Bag 11 slice 94



Labeled Segments Bag 11 slice 94

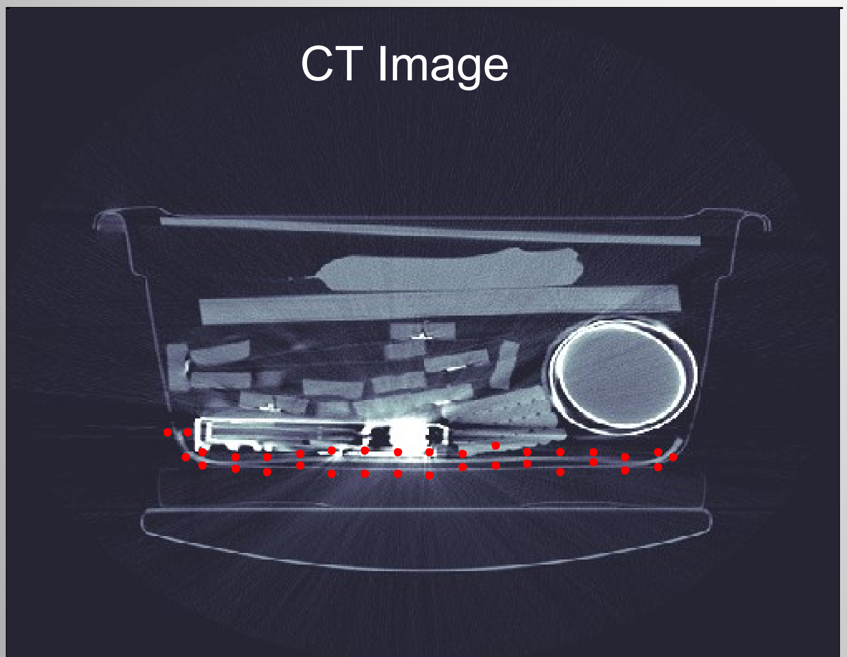


Detected: Yes
Precision:71.9% recall: 44.4%

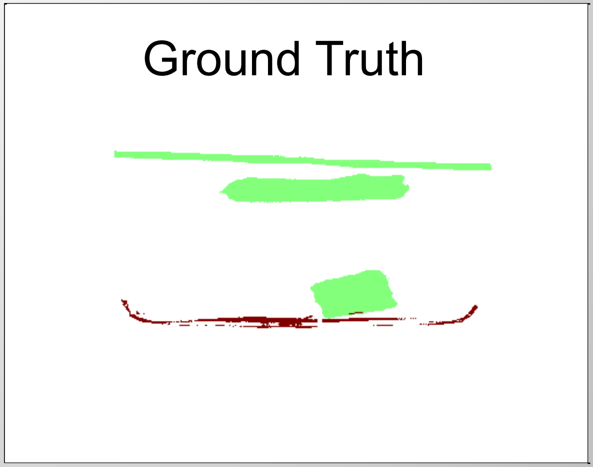
Split into multiple pieces

Case #9: PT sheet based on thickness

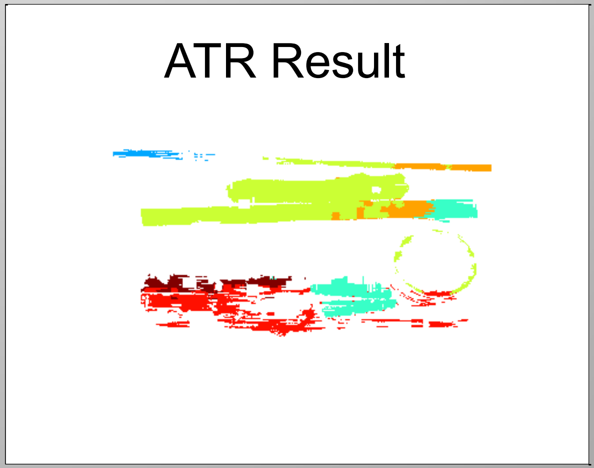
Bag 18 slice 125



Ground Truth Bag 18 slice 125



Labeled Segments Bag 18 slice 125

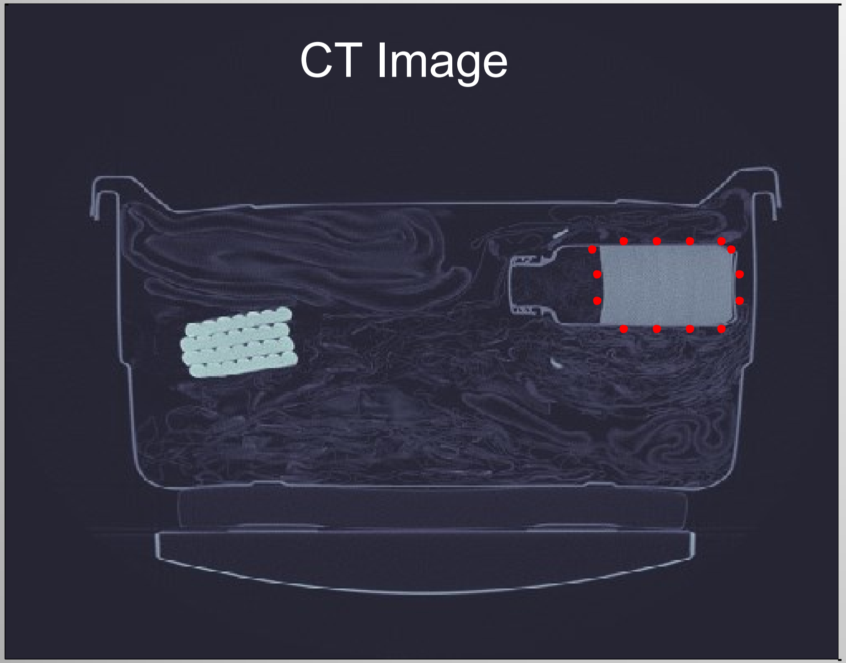


Detected: Yes
Precision: 23.2% recall: 32.6%

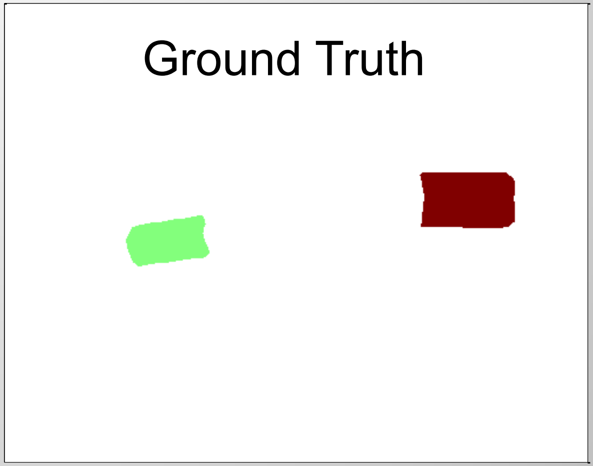
Not well captured and merged with some surroundings

Case #10: PT Powder (based on density, not mass)

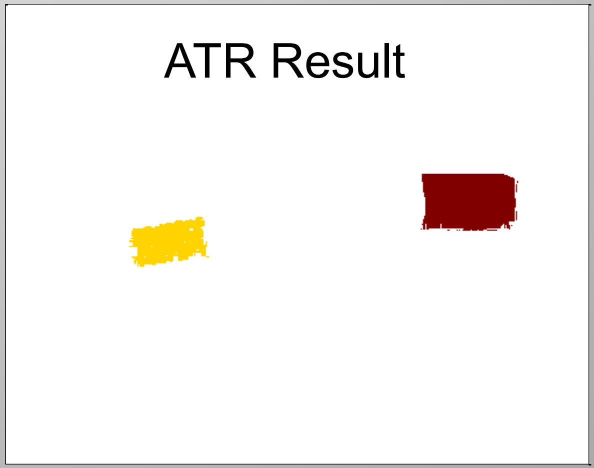
Bag 12 slice 105



Ground Truth Bag 12 slice 105



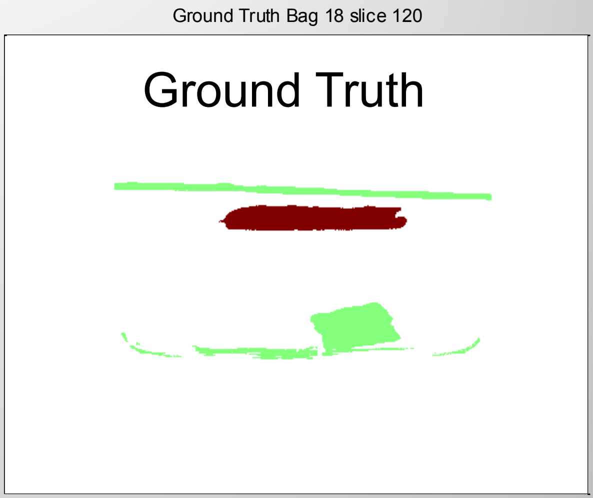
Labeled Segments Bag 12 slice 105

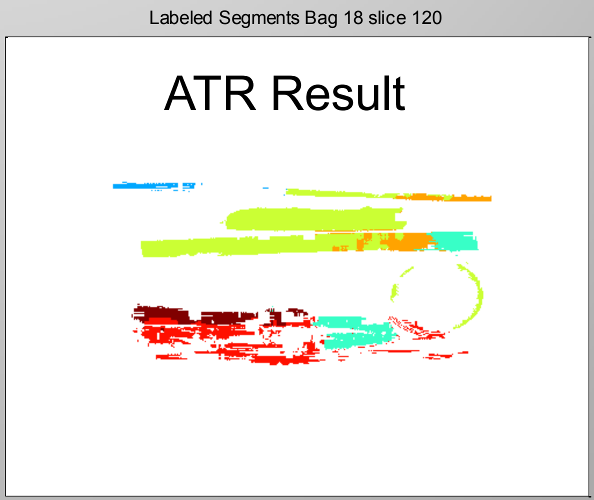


Detected: No
Precision: 49.95% recall: 96.0%

Merged with another object (behind in 3D)

Missed Detection #1: Merger



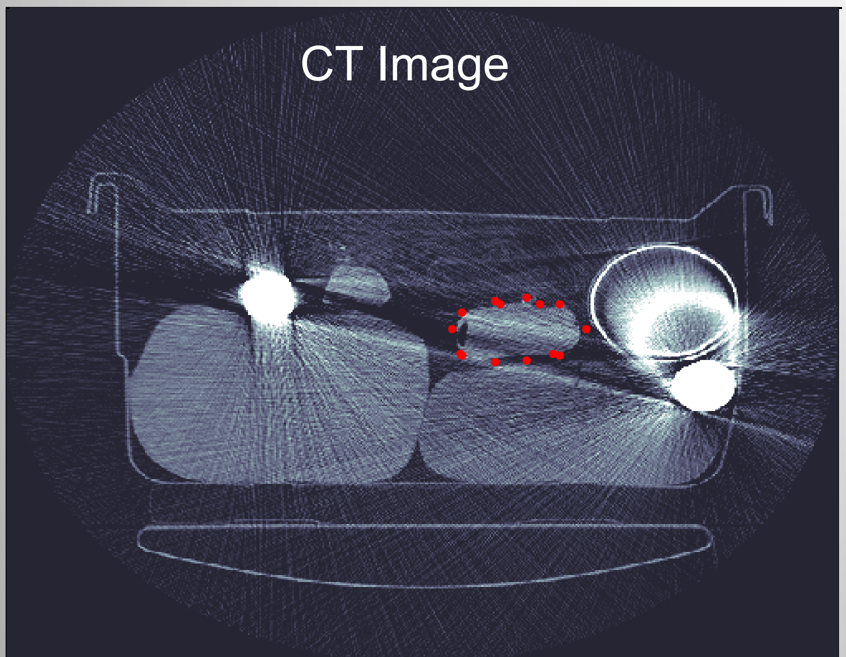


Detected: No
Precision: 28.1% recall: 90.4%

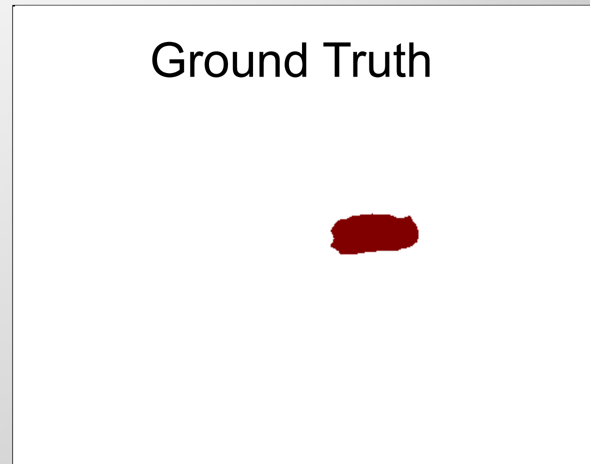
Merged with a large object below it

Missed Detection #2: Metal artifacts

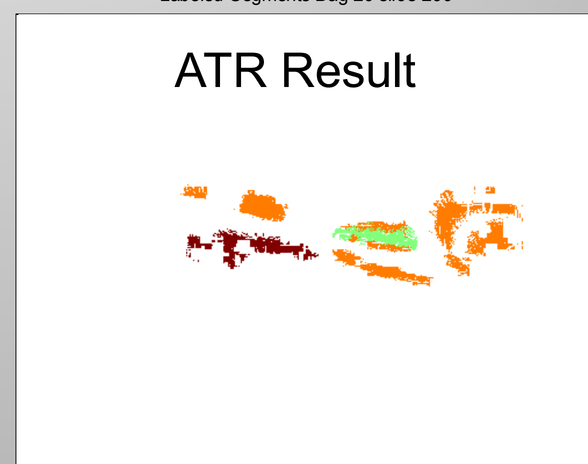
Bag 20 slice 290



Ground Truth Bag 20 slice 290



Labeled Segments Bag 20 slice 290



Detected: No
Precision: 23.0% recall: 38.4%

Really bad distortion and we didn't get
Enough of the object

Discussion

- Strengths – flexibility, robustness to new object characteristics and types
- Weakness – misses some of the targets
  - 7 missed because merged with another object
  - 4 missed due to splitting
  - Bag 18 and Bag 20 split and merged

13 Missed Targets

- 4 were split, 7 were merged with another object
- Bag 18 was split and merged with multiple objects
- Bag 20 was split and merged with a false alarm object

Bag	Target	type	Prec	recall
13	6047	R	92	47
15*	6045	C	98	46
16*	6002	S	33	97
18*	6025	S	28	90
18*	6051	C	79	32
18*	8031	R sh	5	17
20	6012	S	23	38

Bag	Target	type	Prec	recall
34	6012	S	95	43
38	6001	S	41	98
115	6178	S	46	92
147*	6140	R sh	18	65
162*	6573	R sh	15	93
183	6557	S	20	65

* Object detected in some tests but not current best results

Future improvements

- Improve the merge/split algorithms
- Improve sheet processing
- Better final stage classifiers
- Better image reconstruction to reduce streaking artifacts (out of scope for this project), e.g., MAR
- Code refinements
- Apply algorithms to other data sets including potentially classified systems

Comments on the data

- The definition of false alarms creates potentially misleading precision and recall results.
  - A single target split into two parts both of which are detected this situation creates 2 false alarm objects and 1 missed detection
  - Two targets merged together likewise creates 1 false alarm and 2 missed detections

What you learned by participating in this process

- Segmentation is the heart of this problem.
- Algorithm tuning is heavily dependent on the rules of the test
  - We are forced to "tune to the test", which makes the final result less robust to new data whether targets or not. A blind test would be ideal.
- ATR of luggage from CT is hard
  - There is an overlap of targets and non-targets
  - Physics alone is insufficient to get perfection
  - Overtraining is sometimes needed to pass a test – and can lead to 100/0 performance for any known set

Summary of P_D/P_{FA} Results

				No special rules (except for PT sheets)	
Target Type	Target Subtype	Level of Difficulty	Num Targets	Num Detected	PD [%]
Target	All	All	407	381	**93.6**
Target	Clay	All	111	107	96.4
Target	Rubber	All	158	150	94.9
Target	Saline	All	138	124	89.9
Target	Bulk	All	270	251	93
Target	Sheet	All	137	130	94.9
Target	All	Low	77	75	97.4
Target	Clay	Low	29	29	100
Target	Rubber	Low	22	22	100
Target	Saline	Low	26	24	92.3
Target	Bulk	Low	56	54	96.4
Target	Sheet	Low	21	21	100
Target	All	High	317	294	92.7
Target	Clay	High	82	78	95.1
Target	Rubber	High	125	118	94.4
Target	Saline	High	110	98	89.1
Target	Bulk	High	201	185	92
Target	Sheet	High	116	109	94
Pseudo-target	Sheet	High	10	10	**100**
			Num Non-targets	Num FAs	PFA [%]
			1371	163	**11.9**
				Num Scans with FAs	Avg Num FAs
				110	1.57

No special rules:
93.6% / 11.9%

Summary of P_D/P_{FA} Results

				No special rules (except for PT sheets)		New rules added for corner cases	
Target Type	Target Subtype	Level of Difficulty	Num Targets	Num Detected	PD [%]	Num Detected	PD [%]
Target	All	All	407	381	**93.6**	387	**95.1**
Target	Clay	All	111	107	96.4	107	96.4
Target	Rubber	All	158	150	94.9	151	95.6
Target	Saline	All	138	124	89.9	129	93.5
Target	Bulk	All	270	251	93	256	94.8
Target	Sheet	All	137	130	94.9	131	95.6
Target	All	Low	77	75	97.4	77	100
Target	Clay	Low	29	29	100	29	100
Target	Rubber	Low	22	22	100	22	100
Target	Saline	Low	26	24	92.3	26	100
Target	Bulk	Low	56	54	96.4	56	100
Target	Sheet	Low	21	21	100	21	100
Target	All	High	317	294	92.7	298	94
Target	Clay	High	82	78	95.1	78	95.1
Target	Rubber	High	125	118	94.4	119	95.2
Target	Saline	High	110	98	89.1	101	91.8
Target	Bulk	High	201	185	92	188	93.5
Target	Sheet	High	116	109	94	110	94.8
Pseudo-target	Sheet	High	10	10	**100**	10	**100**
			Num Non-targets	Num FAs	PFA [%]	Num FAs	PFA [%]
			1371	163	**11.9**	15	**1.1**
				Num Scans with FAs	Avg Num FAs	Num Scans with FAs	Avg Num FAs
				110	1.57	15	1

No special rules:
93.6% / 11.9%

"Over-trained":
95.1% / 1.1%

Summary of P_D/P_{FA} Results

				No special rules (except for PT sheets)		New rules added for corner cases	
Target Type	Target Subtype	Level of Difficulty	Num Targets	Num Detected	PD [%]	Num Detected	PD [%]
Target	All	All	407	381	**93.6**	387	**95.1**
Target	Clay	All	111	107	96.4	107	96.4
Target	Rubber	All	158	150	94.9	151	95.6
Target	Saline	All	138	124	89.9	129	93.5
Target	Bulk	All	270	251	93	256	94.8
Target	Sheet	All	137	130	94.9	131	95.6
Target	All	Low	77	75	97.4	77	100
Target	Clay	Low	29	29	100	29	100
Target	Rubber	Low	22	22	100	22	100
Target	Saline	Low	26	24	92.3	26	100
Target	Bulk	Low	56	54	96.4	56	100
Target	Sheet	Low	21	21	100	21	100
Target	All	High	317	294	92.7	298	94
Target	Clay	High	82	78	95.1	78	95.1
Target	Rubber	High	125	118	94.4	119	95.2
Target	Saline	High	110	98	89.1	101	91.8
Target	Bulk	High	201	185	92	188	93.5
Target	Sheet	High	116	109	94	110	94.8
Pseudo-target	Sheet	High	10	10	**100**	10	**100**
			Num Non-targets	Num FAs	PFA [%]	Num FAs	PFA [%]
			1371	163	**11.9**	15	**1.1**
				Num Scans with FAs	Avg Num FAs	Num Scans with FAs	Avg Num FAs
				110	1.57	15	1

No special rules:
93.6% / 11.9%

12 new rules
95.1% / 1.1%

Additional rules can lead to
100% / 0%

This is similar to how vendors train to pass the test

Thank you!

- We appreciate being involved in this project with other talented researchers
- Special thanks to Carl, David, Clem and ALERT the team for their guidance and patience
- Laura Parker of DHS for her support of our involvement