

Tabla de Contenido

1. Introducción	1
Contexto	1
El problema de “ <i>Placing</i> ”	4
Objetivos	5
El trabajo realizado.....	5
2. Marco Teórico.....	7
Conceptos importantes	7
El <i>Placing</i> Según Investigadores	10
Herramientas a Utilizar	13
3. Especificación del Problema	19
Tomar y no soltar	19
Variables del <i>placing</i>	19
Acotando el problema	21
Requisitos de Aceptación de Solución	22
4. Descripción de la Solución	23
Trabajo preliminar	23
Descripción General de la Solución	27
Descripción Detallada de Módulos	28
5. Resultados.....	65
Resultados Intermedios	65
Resultados Finales.....	76
Trabajo Futuro	85
Conclusión	89
Bibliografía	90
Anexos	92

Índice de Ilustraciones

<i>Ilustración 1: Robot Roomba</i>	1
<i>Ilustración 2: Ejemplos de navegación autónoma</i>	2
<i>Ilustración 3: Brazo robótico clasificador de items</i>	2
<i>Ilustración 4: Segmentación en escena de mesa con objeto</i>	7
<i>Ilustración 5: Orígenes de sistemas de referencia de los distintos frames de un robot PR2</i>	8
<i>Ilustración 6: Envoltura convexa en un conjunto de datos de 2 dimensiones</i>	8
<i>Ilustración 7: Pose final de un objeto, respecto a frame de referencia</i>	9
<i>Ilustración 8: Etapas de segmentación plana</i>	11
<i>Ilustración 9: Extracto de posicionamiento ejecutado por robot Domo</i>	12
<i>Ilustración 10: Robot El-E asistiendo a paciente con ALS</i>	12
<i>Ilustración 11: Robots que funcionan con ROS</i>	13
<i>Ilustración 12: Pprocesamiento de una escena tridimensional usando librería PCL</i>	15
<i>Ilustración 13: Visualización de Rviz del progreso del robot TurtleBot</i>	15
<i>Ilustración 14: Robot Nao virtual ejecutando tareas de navegación en entorno simulado</i>	16
<i>Ilustración 15: El robot PR2</i>	17
<i>Ilustración 16: Ejemplos de posibles lugares de posicionamiento</i>	20
<i>Ilustración 17: Entorno de trabajo inicial, simulado en Gazebo</i>	24
<i>Ilustración 18: Extracto de archivo de inicialización del robot</i>	29
<i>Ilustración 19: Esquema de perspectivas consideradas para modelamiento</i>	32
<i>Ilustración 20: Gripper con cubo de madera</i>	33
<i>Ilustración 21: Resultado de usar auto-filtro de MoveIt con valores por defecto</i>	34
<i>Ilustración 22: Variando ajustes de auto-filtro</i>	35
<i>Ilustración 23: Visualización de cajas circunscritas al gripper</i>	37
<i>Ilustración 24: Objeto resultante de sustraer cilindro a una esfera</i>	38
<i>Ilustración 25: Caso bidimensional de envoltura convexa</i>	39
<i>Ilustración 26: Objeto amorfo</i>	39
<i>Ilustración 27: Vector normal de un triángulo, posicionado en su centroide</i>	41
<i>Ilustración 28: Centro de masa como descriptor de estabilidad</i>	42
<i>Ilustración 29: Bounding Box de un parche</i>	45
<i>Ilustración 30: Recreación del antes y después de "aplanado" de parche</i>	46
<i>Ilustración 31: Submuestreo de nube de puntos del sensor Kinect</i>	52
<i>Ilustración 32: Nube de puntos desde perspectiva de sensor Kinect</i>	52
<i>Ilustración 33: Desplazamiento de gripper relativo al objeto</i>	57
<i>Ilustración 34: Distintos ángulos válidos de roll para pose final del gripper</i>	58
<i>Ilustración 35: Retirada de gripper tras posicionamiento de objeto</i>	60
<i>Ilustración 36: Mapa organizativo de archivos header de la estructura total integrada</i>	63
<i>Ilustración 37: PR2 sujetando un cubo y un taladro</i>	65
<i>Ilustración 38: PR2 moviendo la cabeza con servicio LookAt</i>	66
<i>Ilustración 39: PR2 moviendo grippers con servicio MoveArm</i>	66
<i>Ilustración 40: PR2 moviendo brazo con un Attached Collision Object esférico</i>	67
<i>Ilustración 41: Gripper scaneado desde nube cruda de puntos de Kinect (sin auto-filtrado)</i>	67
<i>Ilustración 42: Configuración inicial del PR2 para prueba de digitalización de objeto</i>	68
<i>Ilustración 43: Proceso de obtención de modelo geométrico</i>	68
<i>Ilustración 44: Representación del modelo de gripper basado en tres cuboides superpuestas</i>	69

<i>Ilustración 45: Envoltura convexa con normales incorrectamente orientadas..</i>	69
<i>Ilustración 46: Normales correctamente orientadas ..</i>	70
<i>Ilustración 47: Sucesión de procesos aplicados para encontrar pose estable..</i>	71
<i>Ilustración 48: Pose estable detectada para un cubo y para un taladro inalámbrico.</i>	72
<i>Ilustración 49: En color morado, el taladro representado como Collision Object en MoveIt.....</i>	72
<i>Ilustración 50: Planos encontrados por algoritmo de segmentación.....</i>	73
<i>Ilustración 51: Búsqueda de superficies desde distintas ubicaciones..</i>	73
<i>Ilustración 52: Vista de octomap luego de agregar superficie como Collision Object</i>	74
<i>Ilustración 53: PR2 desplazándose hacia superficie encontrada.....</i>	74
<i>Ilustración 54: PR2 desplazándose hacia superficie encontrada.</i>	75
<i>Ilustración 55: Algoritmo de búsqueda de superficie.</i>	76
<i>Ilustración 56: Resultado de búsqueda para distintas posiciones de la mesa.....</i>	77
<i>Ilustración 57: Otras superficies.</i>	77
<i>Ilustración 58: Objetos no detectados como superficie.....</i>	78
<i>Ilustración 59: Búsqueda de superficie entre varias opciones..</i>	78
<i>Ilustración 60: Superficie con objetos encima.....</i>	79
<i>Ilustración 61: Sucesión de detección de pose estable para modelo del cubo..</i>	80
<i>Ilustración 62: Sucesión de cuadros mostrando posicionamiento del cubo.....</i>	80
<i>Ilustración 63: Posicionamiento final del cubo</i>	81
<i>Ilustración 64: Placing de taladro inalámbrico.</i>	81
<i>Ilustración 65: Placing de manilla de puerta..</i>	82
<i>Ilustración 66: Distintos ángulos de "roll" para una misma pose final preliminar del objeto..</i>	82
<i>Ilustración 67: Placing de una manilla de puerta.</i>	83
<i>Ilustración 68: Ejemplo de superficie de posicionamiento muy poblada de objetos.</i>	87
<i>Ilustración 69: Benchmark comparativo de desempeño de planners</i>	93
<i>Ilustración 70: Resultado de auto-filtrado de gripper</i>	95

Índice de Algoritmos

<i>Algoritmo 1: Modelamiento de Objeto</i>	35
<i>Algoritmo 2: Reorientación de normales</i>	41
<i>Algoritmo 3: Búsqueda de parches planos</i>	42
<i>Algoritmo 4: Cálculo de centro de masa estimado</i>	44
<i>Algoritmo 5: Aplanamiento de parche</i>	45
<i>Algoritmo 6: Verificación de intersección entre gripper y plano</i>	46
<i>Algoritmo 7: Búsqueda de pose estable</i>	47
<i>Algoritmo 8: Creación de malla para Collision Object a partir de PolygonMesh de PCL</i>	49
<i>Algoritmo 9: Detección de superficie para posicionamiento de objeto</i>	51
<i>Algoritmo 10: Cálculo de pose para el robot</i>	55
<i>Algoritmo 11: Generación de posibles poses finales para el gripper</i>	59
<i>Algoritmo 12: Testeo de poses y posicionamiento</i>	60
<i>Algoritmo 13: Auto-filtro manual</i>	94