
Índice general

1. Introducción	9
1.1. Visión general	9
1.2. Introducción 1D - Procesamiento de señales sonoras	10
1.3. Introducción 2D - Procesamiento de imágenes	12
1.4. Objetivos de la tesis	12
1.5. Entorno de investigación del doctorando durante el desarrollo de la presente tesis	13
1.6. Estructura de la tesis	15
2. Filtros para aplicaciones de sonido digital	16
2.1. Introducción a los filtros para señales digitales	16
2.2. Tipos de filtro	17
2.2.1. Filtros tipo FIR	17
2.2.2. Filtros tipo IIR	19
2.2.3. Filtros tipo PIIR	20
2.2.4. Comparación entre filtros FIR e IIR	21
2.3. Implementación de los filtros digitales	23
2.3.1. Tecnología ARM [®] NEON [®]	23
2.3.2. NVIDIA <i>Jetson TK1</i>	24
2.3.3. Implementación	25
2.4. Evaluación del rendimiento	28
2.5. Conclusiones	30
3. El Algoritmo de <i>Beamforming</i> (<i>Beamforming</i>)	33
3.1. Introducción	33
3.2. Fundamentos matemáticos del Algoritmo de <i>Beamforming</i>	35
3.3. Implementación del algoritmo de <i>Beamforming Linearly Constrained Minimum Variance (QR-LCMV)</i>	38
3.4. Análisis de resultados previos del algoritmo de <i>Beamforming</i>	39
3.5. Actualización rápida de la factorización QR	45

3.5.1.	La factorización QR	45
3.5.2.	El problema de la actualización QR	46
3.5.3.	El algoritmo de factorización QR por <i>tiles</i>	47
3.5.4.	Modelo de coste para el algoritmo de factorización QR por <i>tiles</i>	50
3.5.5.	El algoritmo de factorización QR por <i>tiles</i> híbrido	52
3.5.6.	Resultados experimentales de la factorización QR	55
3.6.	Un <i>pipeline</i> para la actualización QR en el procesamiento de señales digitales	57
3.6.1.	Descripción del <i>pipeline</i> para actualizar la factorización QR de una matriz	58
3.6.1.1.	Implementación mediante OpenMP	62
3.6.1.2.	Implementación basada en patrones	63
3.6.2.	Resultados experimentales	65
3.6.3.	Evaluación del <i>pipeline</i> con etapas secuenciales	67
3.6.4.	Evaluación del <i>pipeline</i> con etapas paralelas	68
3.7.	El Algoritmo de <i>Beamforming</i> en el NVIDIA <i>Jetson AGX Xavier</i>	69
3.8.	Conclusiones	73
4.	Estimación de profundidad basado en hardware especial	76
4.1.	Introducción	76
4.1.1.	Organización, estructura y definiciones	78
4.1.2.	Descripción de la tecnología	79
4.1.3.	Trabajos relacionados	81
4.2.	Formación de un LF mediante un sistema dual óptico	82
4.3.	Algoritmos de estimación de profundidad	84
4.3.1.	Formas de interpretar un <i>light-field</i> (LF)	84
5.	Algoritmo de rellenado para la estimación de profundidad	93
5.1.	Introducción	93
5.1.1.	Caso práctico de aplicación de mapa denso, reconstrucción 3D	96
5.1.2.	Hardware: Qualcomm Snapdragon	97
5.2.	Algoritmos de aprendizaje automático	98
5.3.	Algoritmo k vecinos más cercanos	99
5.3.1.	Elección del valor correcto de k	100
5.3.2.	Tipos de distancias	101
5.3.3.	Pasos del algoritmo KNN	102
5.3.4.	Técnicas de implementación para búsqueda rápida	102
5.3.5.	<i>Max Heap</i>	103
5.3.6.	<i>kd-tree</i>	105
5.4.	Análisis de la solución adoptada	108
5.5.	Pre-procesos	112
5.5.0.1.	Filtro de outliers	112

5.6. Post-procesos	112
5.6.1. Filtro de mediana	112
5.6.2. Filtro de media	113
5.6.3. <i>Fast Bilateral Solver</i>	113
6. Mapa de profundidad y producto final	115
6.1. Introducción	115
6.2. Presentación del producto	115
6.3. Presentación de los resultados	117
6.3.1. Aplicaciones y resultados obtenidos	117
6.3.2. Comparativa del mapa de profundidad con la competencia	121
6.3.3. Análisis de los resultados	124
7. Contribuciones, conclusiones y trabajo futuro	126
7.1. Introducción	126
7.2. Contribuciones	127
7.3. Notas finales y trabajo futuro	129
Bibliografía	133